



ROBOTSYSTEM FÖR SMÅSERIETILLVERKNING – BÅGSVETSNING

Gunnar Bolmsjö

INSTITUTIONEN FÖR MEKANISK TEKNOLOGI OCH VERKTYGSMASKINER
LUNDS TEKNISKA HÖGSKOLA
1986

ROBOTSYSTEM FÖR
SMÅSERIETILLVERKNING
- BÅGSVETSNING

av

Gunnar Bolmsjö
Civ Ing

Akademisk avhandling som för avläggande av teknologie
doktorsexamen vid Tekniska fakulteten vid Universitetet
i Lund kommer att offentligens försvaras i sal M:B,
Tekniska Högskolan i Lund, fredagen den 25 april, kl 10:15
Fakultetsopponenter: Överingenjör Hans Skoog, ASEA Robotics
AB samt Överingenjör Lars-Göran Eriksson, ESAB AB.

Organization LUND UNIVERSITY Department of Production and Materials Engineering	Document name DOCTORAL DISSERTATION	
	Date of issue April 25, 1986	
	CODEN: LUTMDN/(MMV-1007)1-161/(1986)	
Author(s) Gunnar Bolmsjö	Sponsoring organization National Swedish Board for Technical Development	
Title and subtitle Industrial Robots in Small Series Production - Arc Welding		
Abstract This thesis describes and analyses important factors determining the use of robot systems in small series production, especially for arc welding. To achieve flexibility in small series production the programming time should be reduced to a minimum and the adaptivity should be adjusted in accordance with the movement of the welding gun and the object, to obtain the best welding conditions. A model of a control system with its functions is consequently built up where high level instructions are decomposed to physical movements in the robot system. An object oriented programming technique is described where all welding movements are defined on the object, so that the movement of the welding gun is related to the object. To obtain a welding movement which is adapted and related to the process, a process oriented programming technique is defined. By introducing this technique it is now possible to automatically generate welding movements in two manipulator systems, i.e. robot and positioner. Through the introduction of these new tools a considerably reduced programming time is obtained for the robot system. This is of great importance for the use of welding robots in small series production. The work has revealed that there is a lack of knowledge concerning the control of process parameters. This knowledge is essential to automatically determine the welding parameters and their interrelationships. It is therefor important that further research is directed towards the determination of the relationships between welding parameters and movements to determine a basis for the process programming of a robot system for welding.		
Key words Industrial Robots, Arc Welding, Object Programming, Process Programming, Sensor Interactive		
Classification system and/or index terms (if any)		
Supplementary bibliographical information		Language Swedish
ISSN and key title		ISBN
Recipient's notes	Number of pages	Price
	Security classification	

Distribution by (name and address)

I, the undersigned, being the copyright owner of the abstract of the above-mentioned dissertation, hereby grant to all reference sources permission to publish and disseminate the abstract of the above-mentioned dissertation.

Signature

Gunnar Bolmsjö

Date

19/3 1986



ROBOTSYSTEM FÖR SMÅSERIETILLVERKNING – BÅGSVETSNING

Gunnar Bolmsjö

LUNDS TEKNISKA HÖGSKOLA
1986

Till Siona

FÖRORD

Syftet med denna avhandling är att söka de faktorer som bestämmer användningen av robotsystem vid småserietillverkning och bågsvetsning.

Arbetet har utförts vid institutionen för Mekanisk Teknologi och Verktygsmaskiner vid Lunds Tekniska Högskola.

Arbetet har möjliggjorts genom finansiering av Styrelsen för Teknisk Utveckling (STU) inom ramprogrammet för adaptivt styrda industrirobotar samt utrustning från ESAB AB.

Jag vill i första hand tacka min handledare Professor Gunnar Kullberg som initierat forskningsprojektet och under arbetets gång varit ett ovärderligt stöd samt givit värdefulla industrikontakter.

Ett stort tack framförs också till medarbetare vid institutionen som under arbetets gång bidragit med synpunkter på arbetet.

Till samtliga som bidragit till arbetets genomförande vill jag framföra mitt varma tack.

Lund i mars 1986

Gunnar Bolmsjö

SAMMANFATTNING

Föreliggande arbete är en studie och analys av de faktorer som bestämmer användningen av robotsystem vid småserier och bågsvetsning.

För att åstadkomma flexibilitet vid småserier krävs dels reduktion av programmeringstiden genom styrfunktioner som automatiskt planerar och genererar svetsrörelser, dels adaptivitet genom anpassning av svetsrörelser mot processen.

En modell av ett styrsystem med dessa funktioner har byggts upp där en instruktion genom stegvis förfining bryts ner till fysiska rörelser i robotsystemet.

Vid bågsvetsning med robotar utnyttjas vanligen lägesställare som orienterar och förflyttar objektet så att lämpliga relationer uppstår mellan svetspistol och svetsfog. En objektorienterad programmeringsteknik beskrivs, där alla svetsrörelser definieras utifrån objektet så att svetspistolens rörelser utförs relativt objektet. För att erhålla en svetsrörelse som är anpassad till processen har processprogrammering av rörelsen definierats. Genom att utnyttja denna teknik är det möjligt att automatiskt generera svetsrörelser i tvåmanipulatorsystem.

Användningen av dessa hjälpmedel innebär en kraftigt reducerad programmeringstid. Detta har stor betydelse för användningen av svetsrobotar vid småserier.

Arbetet har emellertid visat att det saknas kunskap om processen för att automatiskt kunna bestämma svetsparametrar och parametrarnas relationer till varandra. Framtida forskning bör därför inriktas mot att bestämma samspelet mellan svetsparametrar och svetsrörelser för att möjliggöra en processprogrammering av svetsrobotsystem.

ABSTRACT

This thesis describes and analyses important factors determining the use of robot systems in small series production especially for arc welding.

To achieve flexibility in small series production the programming time should be reduced to a minimum and the adaptivity should be adjusted in accordance with the movement of the welding gun and the object, to obtain the best welding conditions.

A model of a control system with its functions is consequently built up where high level instructions are decomposed to physical movements in the robot system.

Robotic arc welding is often carried out by incorporating positioners which orient and move objects so that relevant relations are obtained between the welding gun and the weld joint. An object oriented programming technique is described where all welding movements are defined on the object, so that the movement of the welding gun is related to the object. To obtain a welding movement which is adapted and related to the process, a process oriented programming technique is defined. By introducing this technique it is now possible to automatically generate welding movements in two manipulator systems, i.e. robot and positioner.

Through the introduction of these new tools a considerable reduced programming time is obtained for the robot system. This is of great importance for the use of welding robots in small series production.

The work has revealed that there is a lack of knowledge concerning the control of process parameters. This knowledge is essential to automatically determine the welding parameters and their interrelationships. It is therefore important that further research is directed towards the determination of the relationships between welding parameters and movements to determine a basis for the process programming of a robot system for welding.

INNEHALLSFÖRTECKNING

Förord	i
Sammanfattning	ii
Abstract	iii
Förkortningar	vii
 Kapitel 1	 INLEDNING
1.1	Bakgrund 1-1
1.2	Avhandlingens syfte 1-3
1.3	Angreppsätt vid problemlösning 1-3
1.4	Avhandlingens omfattning och avgränsning 1-5
1.5	Avhandlingens disposition 1-7
1.6	Referenser 1-8
 Kapitel 2	 MALBESKRIVNING AV ETT ROBOTSYSTEM FÖR SMASERIETILLVERKNING VID BAGSVETSNING
2.1	Referenser 2-4
 Kapitel 3	 ROBOTSYSTEM OCH DEFINITIONER
3.1	Karakteristiska egenskaper 3-1
3.1.1	Generell uppbyggnad av en industrirobot 3-2
3.1.2	Mekanisk struktur 3-4
3.1.3	Frihetsgrader 3-5
3.1.4	Styrsystem 3-7
3.1.5	Programmering 3-9
3.1.6	Sensorer och sensorsystem 3-11
3.1.7	Integrerade tillverkningssystem 3-12
3.2	Praktisk användning av industriroboten 3-13
3.3	Referenser 3-14
 Kapitel 4	 UTVECKLING AV ROBOTSTATIONER FÖR BAGSVETSNING
4.1	Robotsystem för bågsvetsning 4-4
4.1.1	Mekanisk struktur 4-4

4.1.2	Styrssystem	4-15
4.2	Sensorsystem	4-26
4.2.1	Användning av sensorer och sensorsystem i robotsystem	4-26
4.2.1.1	Bestämning av fel i position och orientering	4-27
4.2.1.2	Bestämning av fel i processen	4-28
4.2.2	Principer för bestämning av tillstånd	4-29
4.2.2.1	Kontaktsensorer	4-29
4.2.2.2	Icke kontaktsensorer	4-30
4.2.3	Beskrivning av olika sensortyper	4-31
4.2.4	Sensorer och sensorsystem för båg- svetsning	4-33
4.2.5	Funktionell uppdelning av sensor- information	4-37
4.2.5.1	Identifiera detalj	4-38
4.2.5.2	Bestämma läge för detalj	4-39
4.2.5.3	Inspektera detalj	4-39
4.2.5.4	Kontrollera manipuleringen	4-40
4.2.5.5	Kontrollera processen	4-41
4.3	Svetsprocessen vid automatiserad svetsning	4-41
4.4	Systemintegration	4-44
4.5	Referenser	4-46
Kapitel 5	UPPBYGGNAD AV ETT STYRSYSTEM FÖR INDUSTRIROBOTAR	5-1
5.1	Realtids styrsystem	5-4
5.1.1	Hierarkisk uppbyggnad av styrsystemet	5-6
5.2	Försöksutrustning för experimentella studier	5-9
5.3	Generering av utdata	5-11
5.3.1	Nivå 1 - Generering av styrdata till rörelseaxlarna	5-16
5.3.1.1	Kroppsrotationen ϕ	5-17
5.3.1.2	Underarmen θ	5-18

5.3.1.3	Överarmen α	5-21
5.3.1.4	Handledsböjningen τ	5-24
5.3.1.5	Handledscrollningen ϕ	5-25
5.3.2	Nivå 2 - Transformerings mellan robot- och kartesiska koordinater	5-26
5.3.2.1	Det direkta kinematiska problemet	5-27
5.3.2.2	Det inversa kinematiska problemet	5-32
5.3.3	Nivå 3 - Generering av enkla robot- rörelser	5-37
5.3.4	Nivå 4 och 5 - Generering av högre funktioner	5-55
5.4	Återkoppling av sensorinformation	5-67
5.5	Målgenerator - robotens världsbild	5-70
5.6	Resultat av systemets funktioner	5-72
5.7	Referenser	5-76
Kapitel 6	ROBOTSYSTEMETS MÖJLIGHETER OCH BEGRÄNSNINGAR VID BAGSVETSNING	6-1
6.1	Referenser	6-5
Kapitel 7	DET PRESENTERADE SYSTEMETS MÖJLIGHETER ATT REALISERAS FÖR PRAKTISK DRIFT	7-1
Kapitel 8	FRAMTIDA FORSKNING	8-1

FÖRKORTNINGAR

A/D	Analog/Digital
AGV	Automated Guided Vehicle
AI	Artificial Intelligence
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CIM	Computer Integrated Manufacture
CP	Contiuous Path
FMS	Flexible Manufacturing System
I/O	Input/Output
MAG	Metal Active Gas
MIG	Metal Inert Gas
PTP	Point To Point
SPM	Synergic Pulsed Mig
TCP	Tool Center Point
TIG	Tungsten Inert Gas

1. INLEDNING

1.1 BAKGRUND

Automation inom industrin har alltid varit ett viktigt medel för att öka produktiviteten, förbättra arbetsmiljön, säkerställa och eventuellt förbättra kvalitén samt reducera kostnaderna. Speciellt har man inom processindustri och massproduktion tidigt kunnat göra stora insatser på automatisering, då en investering kan slås ut över en stor volym eller ett stort antal produkter.

Vid små och medelstora serier har man av ekonomiska och tekniska skäl inte haft samma möjligheter att automatisera produktionen. I produktionen krävs i dessa sammanhang flexibla enheter som har möjlighet att anpassa sig till olika produkter inom vissa variationsområden. Utvecklingen av numeriskt styrda verktygsmaskiner startade på 1950-talet och har inom små- och medelstora serier spelat en mycket stor roll för tillverkningen av detaljer på ett flexibelt och automatiskt sätt. Utvecklingen av maskiner för att hantera arbetsstycken och verktyg kom igång något senare. En ofta använd teknik var att utnyttja pneumatiska enheter som styrdes av programmerbara sekvensnät. Dessa maskiner ligger i gränsområdet av vad man kallar industrirobotar på grund av deras begränsade programmerings- och rörelsemöjligheter. Vid större serier har dessa maskiner kommit till stor användning främst genom sitt låga pris jämfört med servostyrda robotar och sin möjlighet till återanvändning vid annan produktion jämfört med specialanpassade maskiner. De första industrirobotarna togs fram i början på 1960-talet och utnyttjades för miljöovänliga, repetitiva och monotona arbeten.

Vid småserier eller produktionssystem för flexibel tillverkning uppstår problem med ekonomin på grund av krav på kringutrustning och noggrann riggning och inkörning. Om man gör avkall på detta, finns det risk för att processförhållandena kommer att förändras på ett okontrollerbart sätt genom störningar från omgivningen och genom processvariationer. Om förhållandena på arbetsplatsen ändras endast obetydligt efter det att rörelseprogram-

meringen är avslutad, kommer detta med stor sannolikhet att leda till att roboten inte kan utföra den avsedda uppgiften. Det kan t ex röra sig om hantering av detaljer från en palett, där paletten av någon anledning hamnat snett. Detta kommer att leda till att roboten misslyckas med att gripa och manipulera detaljerna om man inte på förhand har rutiner som kan kontrollera palettens läge och orientering. Problemet som uppstår ligger i att den inte är programmerad att lösa en arbetsuppgift. Man har istället definierat ett rörelsemönster som löser den avsedda uppgiften om paletterna och detaljerna har korrekt position och orientering under arbetet.

Om man ser till typ av arbete och vilka krav dessa ställer, kan man dela upp robotarna i

- hanteringsrobotar
- processrobotar
- monteringsrobotar

Hanteringsrobotarna är främst avsedda att manipulera detaljer, t ex överföra detaljer mellan två eller flera transportbanor, maskinbetjäning etc. Processrobotarna utnyttjas för att utföra en process. Detta ställer speciella krav på roboten beroende på vilken typ av process som skall utföras, t ex slipning och punktsvetsning. Vid slipning skall roboten kunna föra en slipskiva med specificerad tryckprofil utefter en friformig yta, medan man vid punktsvetsning bland annat ställer krav på goda accelerations- och retardationsdata. Monteringsrobotarna slutligen ställer mycket höga krav på korta cykeltider och goda kontroll- och styrmöjligheter av periferiutrustningen. Bland dessa robottyper intar bågsvetsroboten en särställning i och med att den inte bara utför och deltar i en process (kontroll och styrning av svetsoperationen) utan även utför ett montage vid svetsningen. Detta innebär att robotsystemet i detta fall inte bara behöver styra och kontrollera processen (ljusbågen eller svetsparametrar) utan även kräver en avkänning av omgivningen via sensorer.

Dagens industrirobotar har den ofullkomligheten att det är ett rörelsemönster som definieras istället för en arbetsuppgift. Detta gör att riggningen eller iordningställandet av en arbetsstation för en tillverkningsserie, där en industrirobot ingår blir tidskrävande. Riggningstiden beror givetvis i stor ut-

sträckning på typ av arbetsuppgift, där enkel materialhantering kräver ringa riggningsarbete, medan processer som bågsvetsning, slipning eller gradning för närvarande kräver betydande riggningsarbeten i fråga om fixturer, verktyg, programmering etc innan produktionen kan starta. Även seriestorleken är intressant då en lång serie lättare kan betala en tidskrävande och dyr riggning än vad småserierna kan. Med ovanstående resonemang kommer man till slutsatsen att man vid mindre seriestorlekar eller vid krav på varierad produktion bör söka sig fram till lösningar av flexibel art, där omriggnings-tiden kan minimeras. Vid större seriestorlekar kan man utnyttja traditionell teknik med detaljbundna fixturer, matare, orienterare samt noggrann inprogrammering och inkörning då kostnaden per detalj ändå kan hållas på en låg nivå.

1.2 AVHANDLINGENS SYFTE

Avhandlingen behandlar robotsystem som kan arbeta mot objekt och process vid bågsvetsning. Detta innebär att robotsystemet anpassar sitt arbetssätt efter de förutsättningar och krav som objekt och process ställer. Härigenom kommer riggningstiden att kunna minimeras genom reducerad programmerings- och inkörningstid. Detta kommer att få stor betydelse för robotanvändningen vid småserietillverkning och tillverkningssystem för kundorderstyrd tillverkning, där man ställer stora krav på snabb omställning av produktionen.

1.3 ANGREPPSSÄTT VID PROBLEMLÖSNING

Dagens industrirobotar har mycket begränsade möjligheter att programmeras för att målinriktat lösa en arbetsuppgift. Med detta menas att roboten skall kunna ges kommandon att utföra en viss uppgift som till utförandet inte nödvändigtvis anges explicit (t ex svetsa plåt A till B, optimera svetsparametrar). För att arbetsuppgiften skall kunna utföras på ett tillfredställande sätt, speciellt då positionsdata eller processparametrar inte längre anges explicit, kommer eventuellt behovet av sensorer att öka. Detta är beroende av hur processen är definierad och hur rörelsernas toleranser kan kontrolleras. Användningen av sensorer i kommersiellt tillgängliga robotsystem begränsas

idag normalt till sådana tillämpningar, där sensorsignaler (oftast binära) mäts vid på förhand bestämda tidpunkter och positioner. Värdet av sensorsignalen bestämmer sedan nästa aktivitet genom villkorliga hopp/tester i applikationsprogrammet. Utnyttjandet av sensorsignalerna är alltså definierat på förhand vilket gör att systemet är att betrakta som stelt, dvs det har mycket liten möjlighet att reagera 'intelligent' vid en icke förutsedd förändring av arbetsförhållandena [1.1].

Om en kontinuerlig återföring av sensorinformation är möjlig, kan robot-systemet lösa sin uppgift även om robotens arbetsförhållanden förändras [1.2]. Roboten kan då i hanteringsexemplet ovan få aktuell information om palettens position och orientering samt hur detaljerna är placerade på paletten. I ett sådant robotsystem är det ej nödvändigt att explicit ange rörelsemönstret för roboten utan en arbetsbeskrivning som

```

hämta detalj A från palett B
placera detalj A i maskin C
starta process D
avsluta process D, placera detalj A i palett E
repetera tills palett B är tom

```

Denna är tillräcklig om systemet har erforderlig information om omgivningen för den aktuella uppgiften. Den typ av information som behövs kommer givetvis att skifta avsevärt mellan olika tillämpningar. Speciellt viktigt är detta då roboten skall medverka aktivt i processer av olika slag, t ex slipning, svetsning, etc, där parametrar som har med själva processen att göra skall övervakas och styras samt integreras med robotens arbete.

För att kunna ändra robotsystemets arbetssätt under processen krävs sensorer som ger styrsystemet information om processen och algoritmer som utifrån denna information återkopplar en rörelse som är anpassad till processen. Det kan gälla att identifiera paletter eller detaljer, bestämma position och orientering med tillräcklig noggrannhet, kvalitetskontroll i arbetsprocessen, etc. För att kunna arbeta och generera rörelser som är anpassade till objekt och process krävs en rörelsegenerering som är processororienterad. Detta innebär t ex vid bågsvetsning att kontrollera samspelet mellan fogens

orientering, svetspistolens orientering och robotens arbetsområde. Om dessa aktiviteter skall ske under arbetsprocessen, krävs relativt stor datakraft. Normalt eftersträvar man att roboten skall arbeta minst lika fort som en människa. Detta medför att sensorsignaler på låg nivå (banstyrning) skall resultera i en omedelbar reaktion (typiskt millisekunder). Sensorsignaler på högre nivå (beslutsval, strategi i arbetsprocessen) som inte innebär en direkt fysisk reaktion, kan ha en längre responstid (typiskt sekunder) eftersom utsignalen påverkar robotens arbete i kommande moment. Detta kan exempelvis gälla planering av den samordnade rörelsen i ett tvåmanipulatorsystem vid bågsvetsning. Den mer strategiska planeringen av arbetsstationen är en sådan verksamhet som kan fortgå under arbetsförloppet och är inte av realtidskaraktär. Då roboten arbetar i en process (slipning, svetsning, limning, etc) är processkännedom av stor betydelse då igenkänning av ett speciellt tillstånd kan kräva speciell kunskap eller speciella beslut/sensorer för att kunna tas om hand på ett korrekt sätt. Ett korrekt utnyttjande av ett styrsystem som har möjlighet att utnyttja en mängd sensordata och processdata för att styra robotens arbete gör det möjligt att arbeta med korta serier inom de ramar som ges av robotens geometri, lägesställare och fixturer. En sådan användning av systemet leder till begreppet Flexible Manufacturing Systems (FMS) där man eftersträvar ett tillverkningssystem som tillåter varierad och blandad produktion (i detta fall för svetsning) i små serier. Begreppet FMS kommer att definieras i kapitel 3 under 'Integrerade tillverkningssystem'.

1.4 AVHANDLINGENS OMFATTNING OCH AVGRÄNSNING

Avhandlingen beskriver en ny modell över ett robotsystem för svetsning av småserier. Flera faktorer måste samverka i ett sådant system för att tillräcklig hög flexibilitet skall uppnås:

1. Adaptivitet mot omvärlden genom att under svetsning möjliggöra fögföljning samt automatiskt val och reglering av svetsparametrar.
2. Reduktion av programmeringstiden. Detta kan åstadkommas genom att programutvecklingshjälpmedel (CAD/CAM, Off line, On line) tas fram för olika situationer i tillverkningsprocessen som t ex samordnade rörelser i ett flermanipulatorsystem och automatisk generering av svetsrörelser.

Delar av ett sådant system har byggts upp för att experimentellt verifiera gjorda antaganden. Följande enheter i robotsystemet diskuteras i avhandlingen:

- Kinematik med robotarm och lägesställare
- Styrsystem
- Sensorsystem
- Svetsutrustning

Robotarmens rörelser kontra lägesställarens behandlas så att man erhåller en synkroniserad rörelse av arbetspunkten mot ett rörligt objekt som är fastsatt på en lägesställare.

Styrsystemuppbyggnaden diskuteras med speciell hänsyn till sensoråterkoppling i realtid och aspekter på olika programutvecklingshjälpmedel.

Sensorsystem och metoder som återkopplar information om svetsprocessen diskuteras.

Svetsutrustningens påverkan på svetsresultatet och möjligheter till styrning och automatiskt val av svetsparametrar under svetsning behandlas. Detta är av stor betydelse för snabb inställning av korrekta svetsparametrar vid småserier.

1.5 AVHANDLINGENS DISPOSITION

Avhandlingen behandlar speciella krav som ställs på en problemlösande svetsrobot och hur den skall arbeta tillsammans med periferiutrustningen för att man skall få en god produktivitet vid småserier. Ett koncept beskrivs för uppbyggnad av ett styrsystem som har möjlighet att välja och utföra samordnade rörelser av robot och lägesställare samt kunna återkopplas med sensordata i realtid.

Kapitel 2 tar upp en krav- och målspecifikation av ett robotsvetssystem för småserietillverkning vid bågsvetsning.

Kapitel 3 behandlar definitioner kring begreppet industrirobotar och beskriver dess egenskaper och användning vid bågsvetsning.

Kapitel 4 ger en översikt över forskning och utveckling inom svetsrobotar och styrningen av dessa för bågsvetsning och småserier.

Kapitel 5 beskriver en modell över hur ett styrsystem kan byggas upp innehållande högre styrfunktioner samt realtids återkoppling för fögföljning.

Kapitel 6 behandlar de möjligheter och begränsningar som sätts av svetsprocessen för robotiserad bågsvetsning.

Kapitel 7 diskuterar det presenterade systemets möjligheter att realiseras för praktisk användning vid robotiserad bågsvetsning.

Kapitel 8 diskuterar framtida forskning inom området.

1.6 REFERENSER

- 1.1 H. R. Holt, Robot Decision Making, SME Technical Paper MS 77-751, Dearborn Michigan (1977).
- 1.2 D. Nitzan, Robotic Sensors in Programmable Automation, Technical Note 183 March 1979, SRI International, California, USA.

2. MALBESKRIVNING AV ETT ROBOTSYSYSTEM FÖR SMASERIETILLVERKNING VID BAGSVETSNING

Med utvecklingen av ett robotsystem för småserietillverkning vid bågsvetsning följer en mängd unika krav på egenskaper som vanligtvis inte uppfylls vid en robotstation för bågsvetsning. Krav man normalt ställer på robotsystem kommer inte att behandlas här såsom t ex nödstoppsrutiner för att garantera säkerhetskrav, krav på speciell periferiutrustning för in/ut-transport av arbetsmaterial eller de organisatoriska krav som ställs på robotsystemet för att ett effektivt utnyttjande skall uppnås vid småserier.

Småserier innebär en användning vid arbetsuppgifter som är av flexibel karaktär. Med flexibel menas här tillverkning av en varierad produktflora inom ett begränsat variationsområde (geometri, dimension, komplexitet). Detta innebär att robotsystemet främst skall användas, där konventionell programmeringsteknik är tidsödande. Styrsystemet skall kunna anpassa sitt arbetssätt under svetsning med avseende på inhämtade sensorsignaler (ändrade arbetsförhållande) samt kunna göra ett snabbt och korrekt val av svetsparametrar. Robotsystemet skall kunna byggas ut med nya funktioner som är anpassade till nya arbetssituationer (styralgoritmer, sensoråterkopplingar, kommandon). Nedan följer en kravspecifikation på de speciella egenskaper som gäller för ett robotsystem för bågsvetsning:

- a) Systemet skall kunna hämta och processa sensordata från omvärlden (svetsprocessen och andra relevanta arbetsförhållanden) och utifrån denna information ändra robotens arbetssätt på ett målinriktat sätt så att eventuella fel minimeras till en tillåten toleransnivå [2.1].

Styrsystemet skall under arbetsförloppet kunna ändra robotsystemets arbetssätt (robot, lägesställare, svetsmaskin) utifrån de data om arbetsförhållanden som sensorerna ger. De korrigeringar som utförs i arbetssättet skall ske med tillräcklig täthet i tiden,

beroende på hur snabba förändringar som kan förväntas vid en given process. Vid följning och en svets hastighet av 10 mm/s är uppdatering av rörelsen var 10:e ms nödvändig för att man skall ligga i området av robotens egen rörelseupplösning (0.1 mm). Andra sensordata kan gälla svetsparametrar eller fixturstatus. Vid alla korrigeringar i arbetssätt hos systemet är det viktigt att dessa kan ske på ett målinriktat sätt för att reducera de uppmätta felen.

- b) Systemet skall kunna programmeras med ett målinriktat och kraftfullt kommandospråk på olika nivåer så att programmeringstiden vid robot-systemet blir så kort som möjligt [2.2].

För applikationer som inbegriper småserier och bågsvetsning är programmeringstiden vid roboten av stor betydelse för stationens lönsamhet. Det är därför av stor vikt att systemet kan programmeras med off line programmeringsteknik. Kommandospråket skall vara av målinriktad karaktär, där rörelser eller systemfunktioner inte alltid skall behöva anges explicit. Det skall vidare i så stor utsträckning som möjligt vara oberoende av den kinematiska konfigurationen i systemet, vara interaktivt samt medge simulering och editering av program. Systemet skall förutom off line programmering även medge en on line programmering för intrimning och korrigering av program gjorda med off line programmeringsteknik.

- c) Systemet skall kunna kommunicera med andra system och externa databaser som hjälp vid programmering och val av processdata [2.3].

Vid off line programmeringstekniken är det viktigt att kunna utnyttja redan existerande databaser, där detaljer, relationer mellan detaljer, processdata etc är definierade (t ex CAD/CAM databaser och svetsparameterval). Rörelsen eller processdata behöver då inte anges explicit, utan dessa kan beräknas av en dator utifrån de förhållanden som har definierats vid arbetsstationen, t ex

en rörelse utifrån detaljer som är inplacerade i ett robotsystem och dessas aktuella koordinatsystem. Ett annat viktigt fall, där externa databaser eller system kan utnyttjas, är styrning och val av svetsparametrar för en på förhand definierad fogtyp.

- d) Systemet skall vara enkelt utbyggbart för nya funktioner och ändrade arbetsförhållanden [2.4].

Robotsystemet är avsett för flexibel automation vid bågsvetsning och det är därför svårt att förutsäga framtida användning och behov av olika funktioner på grund av den variationsrikedom som kan förekomma hos produkter i fråga om komplexitet, dimensioner, material, etc. Robotsystemet och dess styrsystem måste därför ha en sådan uppbyggnad att det är enkelt att utföra ändringar i kommandospråket, lägga in nya återkopplingsrutiner för sensorstyrning, ändra transformationsrutiner för modifierad kinematik, etc. I och med att styrsystemet byggs upp med en sådan struktur vinner man dessutom ett relativt oberoende av hur den mekaniska strukturen ser ut, då man vid förändring av kinematiken endast behöver ändra vissa delar som transformationsrutiner och interpolationsrutiner.

- e) Styrsystemet till robotsystemet skall i största utsträckning vara oberoende av olika mekaniska strukturer (robot, lägesställare) och enkelt att anpassa till dessa [2.4].

Styrsystemet för robotsystemet skall ha sådana egenskaper att det utan alltför stora förändringar eller arbetsinsatser kan användas till olika mekaniska strukturer. Detta gör att likartade styrsystem med samma typ av kommandospråk kan användas vid olika robotsystem vilket ger fördelar i erfarenhet, säkerhet, programmering och handhavande. På sensorsidan vinner man teknologiska erfarenheter genom likartade sensorarrangemang och återkopplingsstrategier vid liknande situationer på olika robotsystem. Styrsystemets uppbygg-

nad skall vara sådan att en kontrollerad rörelse med TCP på roboten mot ett rörligt arbetsstycke fastsatt på lägesställaren skall kunna utföras.

- f) Robotsystemet skall automatiskt kunna styra svetsprocessen till ett fullgott resultat [2.5].

Vid småserieproduktion krävs en snabb inställning av svetsparametrar för de aktuella svetsfogarna. Robotsystemet måste därför självt utifrån utgångsdata som fogtyp, material, dimensioner, etc, kunna välja och styra svetsparametrar integrerade med svetsrörelsen för att kunna uppnå ett fullgott resultat vid småserier. Oavsett vilka resurser och egenskaper som ett robotsystem har beror det dock ytterst på användaren hur effektivt och flexibelt systemet blir i verklig drift.

2.1 REFERENSER

- 2.1 J. J. Solberg, R. P. Paul and D. C. Anderson, The Factory of the Future: A Framework for Research, Computer Based Factory Automation, Proc 11th Conf on Production Research and Technology, 1984, Pittsburgh, Pennsylvania, USA.
- 2.2 M. Parent and C. Laugeau, Logic and Programming (Robot Technology Volume 5), pp 113-116, Kogan Page, UK, 1984, (Studentlitteratur, Lund, in Scandinavia).
- 2.3 S. B. Jones and B. Koch, The Application of CIM to Welder Fabrication, Automation and Robotisation in Welding and Allied Processes, Proc of the Conf, Strasbourg, France, 1985.
- 2.4 A. J. Barbera, J. S. Albus and M. L. Fitzgerald, Hierarchical Control of Robots Using Microcomputers, Proc of the 9th Int Symp on Industrial Robots, Washington D.C, USA, 1979.
- 2.5 J. Kremers et al, Development of a Machine-Vision-Based Robotic Arc-Welding System, Proc of the 13th Int Symp on Industrial Robots, Chicago, USA, 1983.

3 ROBOTSYSTEM OCH DEFINITIONER

3.1 KARAKTERISTISKA EGENSKAPER

En definition av en industrirobot och dess egenskaper gjordes 1979 av RIA (Robot Institute of America) säger att [3.1]

"A robot is a reprogrammable multifunctional manipulator designed to move material, parts, tools or specialized devices, through variable programmed motions for the performance of a variety of tasks."

En fri tolkning av denna definition ger att en maskin som kan klassas som en robot skall ha minne och logik för att kunna arbeta självständigt och automatiskt samt ha en sådan mekanisk uppbyggnad att den kan användas för ett flertal uppgifter. En vidareutveckling av definitionsbegreppet visar att det främst är två karakteristiska egenskaper som krävs av en industrirobot, nämligen mångsidighet och adaptivitet mot omvärlden. Anledningen till dessa utökade krav är att roboten är en del av ett produktionssystem och måste anpassas till de arbetsuppgifter produktionssystemet skall lösa (robot-svetsning vid småserier i detta fall).

Mångsidighet innebär att roboten skall ha en mekanisk uppbyggnad som gör att den kan utföra olika uppgifter och eventuellt kunna utföra samma uppgift på olika sätt. Detta innebär även att den skall kunna återanvändas vid förändringar i produktionen.

Adaptivitet mot omvärlden innebär att roboten skall kunna utföra en uppgift av sig själv oavsett normala förändringar som uppstår på arbetsplatsen. Denna egenskap kräver att roboten är försedd med ett sensorsystem som kontrollerar

sin omgivning hela tiden under arbetet. Ställer man kravet på denna nivå finns det troligen ingen robot idag som har denna egenskap. Vid många applikationer finns det emellertid möjlighet att definiera robotens arbetsförhållanden så att man till tid och rum vet var förändringarna kan uppstå. I dessa fall har en hel del av dagens robotar även denna egenskap, trots att robotens sinnen är mycket begränsade jämfört med människans möjligheter att bestämma och analysera sin omgivning.

I begreppet adaptivitet mot omvärlden ligger de egenskaper som krävs för att roboten skall kunna användas vid småserieproduktion. Vad som avses i texten med olika seriestorlekar definieras av följande uppställning:

enstyck	1-	10
småserier	11-	500
medelstora serier	501-	10000
stora serier	10001-	500000
massproduktion	500001-	

Vid robotsvetsning och småserier är det inte enbart seriestorleken som är av intresse. Man måste också ta med i beräkningen, om tillverkningen är återkommande för att bedöma de ekonomiska möjligheterna vid en tillverkning.

3.1.1 GENERELL UPPBYGGNAD AV EN INDUSTRIROBOT

En industrirobot består i det allmänna fallet av ett flertal komponenter som samverkar med varandra. Robotens styrning sköts av ett styrsystem som utifrån ett väldefinierat språk beskriver hur roboten skall arbeta för en given uppgift. För att den skall kunna utföra styrkommandon krävs styrprogram som kan beräkna banor och styra manipulatorns enskilda leder så att man får en önskad rörelse av en arbetspunkt, ofta kallad Tool Center Point eller TCP. För denna beräkning krävs interpolations- och transformationsrutiner som kan beräkna hur manipulatorn skall genomlöpa den aktuella banan uttryckt i kartesiska koordinater över till robotkoordinater och sedan vidare till

styrsignaler för drivdonen på manipulatorns leder. För att roboten skall kunna utföra önskad uppgift krävs normalt en samverkan mellan robotsystemet och omvärlden. Detta möjliggöres genom digitala eller analoga in- och utgångar. Med utnyttjande av dessa in- och utgångar samt möjligheten att utföra logiska operationer (villkorstester, välja alternativ) på informationen från omvärlden har styrsystemet möjlighet att i givna situationer kontrollera och styra sin verksamhet.

Drivdonen ger under kontrollerade former kraft till manipulatorn. Drivdonen kan vara av elektrisk, hydraulisk eller pneumatisk typ. I de flesta fall eftersträvar man att placera drivdonen i robotens kropp för att inte reducera möjlig lastkapacitet. I sådana fall krävs en transmission mellan drivdon och den rörelseaxel som drivdonet skall påverka. Transmissionens uppgift är förutom att kunna överföra kraft över avstånd, även att ge utgående moment och hastighet anpassad till aktuellt drivdon. Exempel på transmissioner som används är länkar, remmar, kuggtransmissioner och kulmutterskruvar.

Den mekaniska strukturen och kinematiken består av armar förbundna med varandra i vanligen enkla translations- eller rotationsleder och har till uppgift att upprätthålla korrekta förhållanden mellan arbetsdon (gripdon, verktyg) och objekt (det objekt roboten arbetar mot, t ex arbetsstycke eller maskin).

Slutligen finns det sensorer som skall kunna ge styrsystemet information dels om roboten själv (interna sensorer), dels om robotens omvärld (externa sensorer). Interna sensorer används huvudsakligen för att mäta position, hastighet och acceleration samt olika larmnivåer (nödstopp, strömmar, etc) på roboten. Externa sensorer används för att roboten skall kunna ändra sitt förutbestämda arbetssätt utifrån händelser i omgivningen.

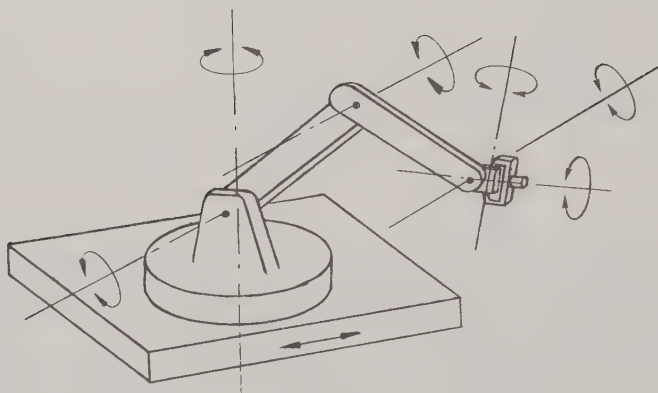


Bild 3.1 Mekanisk struktur hos en industrirobot med åkdon, arm och handled.

3.1.2 MEKANISK STRUKTUR

Den mekaniska strukturen består av ett antal armar sammankopplade med translations- eller rotationsleder. Man kan funktionellt dela upp strukturen i tre samarbetande delar, åkdon, arm och handled.

Åkdonet har till uppgift att förflytta roboten till arbetsområdet. Armen skall sedan förflytta handleden till en position i arbetsområdet så att man uppnår korrekt TCP då handleden har den önskade orienteringen. Detta kan illustreras som i bild 3.1. Även om det teoretiskt är möjligt att ge åkvagnen sex frihetsgrader (eller oberoende rörelsemöjligheter), är det normala då åkdon används att utnyttja en- eller tvåaxliga åkdon beroende på de krav som applikationen ställer.

Då armens uppgift är att positionera handleden till en bestämd position i rummet krävs det tre oberoende frihetsgrader för att klara detta. Som nämnts tidigare utnyttjar man en kombination av translations- och/eller rotationsaxlar för att åstadkomma armens olika rörelser. Genom att kombinera dessa är det med tre rörelseaxlar möjligt att erhålla 37 olika mekaniska armstrukturer som sveper en volym [3.2]. Emellertid har endast fem av dessa möjliga typer

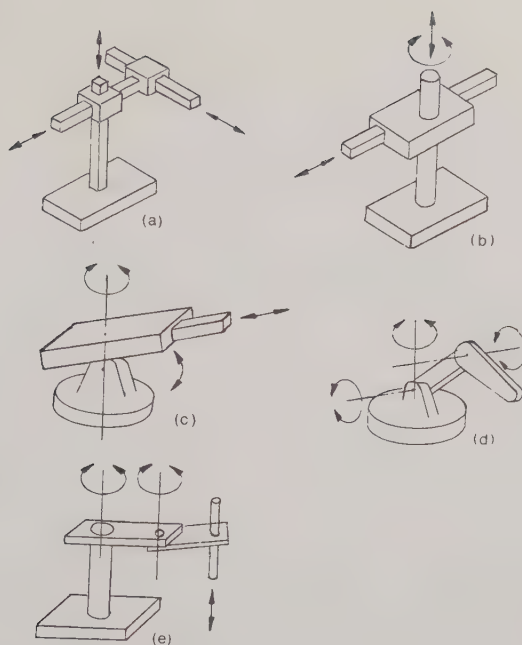


Bild 3.2 Armstrukturer i praktisk användning. (a) kartesisk, (b) cylindrisk, (c) sfärisk, (d) revolut, (e) SCARA (Selective Compliance Assembly Robot Arm, selektivt eftergivlig).

kommit till praktisk användning, se bild 3.2.

3.1.3 FRIHETSGRADER

Ett objekt som relateras till ett ortogonalt koordinatsystem Ω kan utföra tre oberoende translationer och tre oberoende rotationer, dvs totalt sex oberoende rörelser. Man säger att objektet har sex frihetsgrader relativt Ω , se bild 3.3. Om ett förhållande skapas mellan två objekt, förlorar objekten frihetsgrader i förhållande till varandra. Detta betyder att om roboten placeras i systemet Ω , måste roboten ha sex oberoende frihetsgrader för att helt "ta över" ett fritt objekts frihetsgrader. Om objektet har en viss symmetri, kan ett färre antal frihetsgrader vara tillräckligt för att godtyckligt orientera och positionera ett gripdon mot objektet (att gripa en sfär kräver t ex endast tre frihetsgrader).

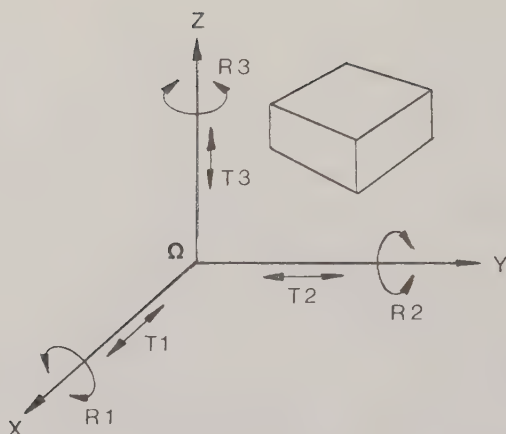


Bild 3.3 Antalet frihetsgrader för ett objekt i ett ortogonalt koordinatsystem Ω .

Även om man teoretiskt skulle kunna klara sig med sex frihetsgrader för att komma i godtycklig position och orientering, har man i praktiken ofta användning för ytterligare rörelseaxlar. Antalet oberoende frihetsgrader eller axlar kan aldrig bli fler än sex. Om åkdon används, ger detta extra rörelseaxlar. Andra exempel är när man har problem med hinder i arbetsområdet och förser robotarmen med extra rörelseaxlar för att trots detta kunna komma i rätt läge för arbetet (t ex sprutmålning), se bild 3.4a, eller där man låter arbetsstycket få extra rörelseaxlar för att dels uppnå god åtkomlighet, dels uppnå goda förutsättningar för arbetsprocessen (t ex vid bågsvetsning för att kunna manipulera objektet så att man för processen kan upprätthålla rätt fogläge och pistolvinkel), se bild 3.4b.

Vid flera än sex rörelseaxlar i systemet får man ett överbestämt styrproblem och de "överflödiga" rörelseaxlarnas rörelser får bestämmas via bivillkor som ställs av processen eller arbetsplatsen. Detta kommer att diskuteras ingående längre fram i samband med styrning av robotsvetsstation med manipulator och lägesställare om totalt sju rörelseaxlar.

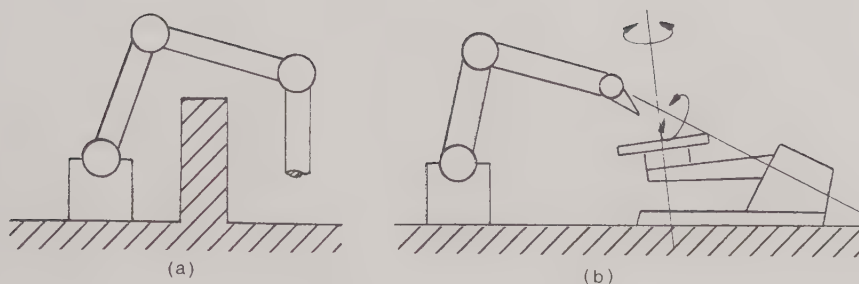


Bild 3.4 Exempel på användning av extra rörelseaxlar.

3.1.4 STYRSYSTEM

Robotens möjlighet att utföra olika typer av uppgifter inom de gränser som mekaniken ger, sätts till stor del av styrsystemet. Man kan urskilja två huvudprinciper för styrning av industrirobotar, Point-To-Point-styrning (PTP) och Continuous Path-styrning (CP). Fram till slutet av 1970-talet var nästan samtliga styrsystem av PTP-typ men i och med en fortsatt utveckling inom mikroprocessortekniken har det på senare tid blivit möjligt att utveckla CP-system.

PTP-styrning styr robotens rörelseaxlar till inprogrammerade positioner i robotens rörelserymd utan att rörelsen mellan punkterna är känd på förhand. Rörelsen kan interpoleras fram osynkroniserat (robotens rörelseaxlar når slutpositionen individuellt vid olika tidpunkter) eller synkroniserat (robotens rörelseaxlar når slutpositionen samtidigt). I bägge fallen utför roboten sin rörelse i robotkoordinater (ett koordinatsystem som är bestämt av robotens mekaniska uppbyggnad).

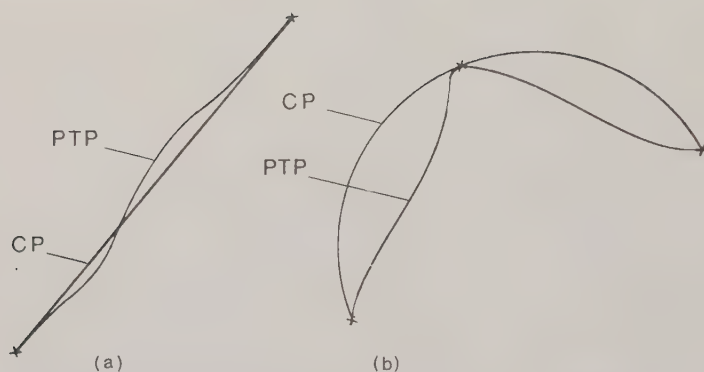


Bild 3.5 Skillnad i rörelse mellan PTP-styrdd och CP-styrdd robot.

Vid CP-styrning har man förutom slutpositionen även möjlighet att definiera rörelsen i ett kartesiskt koordinatsystem till bana och hastighet, t ex linjär, se bild 3.5a, eller cirkulär, se bild 3.5b. CP-styrning kräver en avancerad datorstyrning då man vid interpolerande system förutom själva interpoleringen av banan i det kartesiska koordinatsystemet måste transformera varje interpolerad punkt till robotkoordinater, dvs beräkna hur robotens rörelseaxlar skall positioneras för att uppnå önskad rörelse. En speciell typ av CP-styrning uppvisar de s k play-back robotarna, där man vid programmeringen för roboten i det önskade rörelsemönstret och samtidigt spelar in axelpositionerna i ett minne. I och med detta krävs det ingen avancerad beräkning för att få roboten att röra sig i den önskade banan, utan robotsystemet spelar helt enkelt upp rörelsen punkt för punkt och erhåller på detta sätt en kontinuerlig rörelse.

Förutom rörelsestyrning krävs en logisk styrning av sekvenser som gör det möjligt för roboten att arbeta mot en föränderlig omgivning. I det enklaste fallet gäller det att styra omgivningen genom digitala styrsignaler. För att synkronisera och/eller kunna anpassa roboten mot händelser i omgivningen har man också möjlighet att ta in signaler i systemet. Normalt kan man i programflödet göra enkla tester på dessa signaler, eventuellt utföra heltalsaritmetik på indata och sedan utifrån resultatet utföra någon fysisk aktivitet, t ex

anropa ett underprogram eller styra en utsignal. Denna del av styrsystemet betecknas I/O-snitt (In/Out) eller sensor-snitt. De styrsignaler som används arbetar vanligen med likspänningar, där 24V betecknar en etta och 0V en nolla. För vissa tillämpningar eller sensorer krävs möjlighet att ta in mera information än vad den rena binära informationen kan ge. Man utnyttjar då A/D-omvandlare (Analog/Digital) som överför en analog spänning till ett digitalt siffervärde med önskad upplösning. Beroende på styrsystemets sofistikeringsgrad har man sedan möjlighet att styra och kontrollera robotens omgivning via I/O-snittet.

3.1.5 PROGRAMMERING

Programmeringen av de flesta robotar som finns på marknaden idag sker på själva robotsystemet genom att man har kopplat en programmeringslåda till styrsystemet som styr programinmatningen via t ex menyer. Menyerna anger tillåtna kommandon och underlättar på så sätt programmeringen genom att man inte behöver komma ihåg en stor mängd instruktioner. Programmeringen av rörelsemönster på dessa system kan ske på två sätt, learning-by-showing eller learning-by-teaching. Learning-by-showing innebär att man tar tag i robotarmen och för den i det rörelsemönster som man vill att den skall utföra. Learning-by-teaching innebär att man lär roboten ett rörelsemönster via en styrspak eller styrknappar och önskad position sparas genom ett speciellt lagringskommando. Den teknik som är beskriven ovan brukar kallas för on line programmering, eftersom man använder robotens programmeringssystem (styrsystem och manipulator) under själva programmeringen.

Under senare år har det för vissa robotar börjat komma fram möjligheter att programera dem off line. Med off line programmering menas att programmeringen av arbetsstationen huvudsakligen utförs på någon annan dator (programmeringssystem) för att i ett senare skede överföras till robotstationen via kabel eller annat medium som diskett eller band. Inom off line programmeringstekniken finns tre filosofier, robotorienterad, objektorienterad och processororienterad programmering.

Vid programmering med ett robotorienterat språk krävs rörelseorienterande/logikorienterande kommandon för varje handlingssteg med numeriska data, t ex som i programspråket VAL [3.3]

MOVE punkt
WAIT tid
SIGNAL kanal

Det objektorienterade språket behandlar de objekt som skall manipuleras och deras relationer till varandra, t ex som i programspråket AUTOPASS [3.4]

PLACE interlock ON bracket SUCH THAT interlock.hole
IS ALIGNED WITH bracket.hole AND interlock.base
CONTACTS bracket.top

Det processororienterade språket är av en problemlösande natur och utnyttjar kunskap om de produkter som skall manipuleras samt deras relation till processen - robotstationen. Detta kan enligt exemplet ovan se ut som

ASSEMBLE interlock ON bracket

Nästa steg i utvecklingen är att söka utnyttja CAD/CAM-tekniken (Computer Aided Design/Computer Aided Manufacturing) för programmering av robotens rörelser för att på så sätt få en enklare definiering av rörelsemönster utifrån en definierad modell av omgivningen.

Ett väsentligt problem som uppträder vid övergång till off line programmering är robotens och objektets absoluta noggrannhet. Vid traditionell programmering lär man roboten en position genom att förflytta den till ett läge utifrån en verklig modell (arbetsstycke) och har då endast repeternoggrannheten att ta hänsyn till. Vid off line programmering räknar man fram en position utifrån en matematisk modell och får på så sätt in felkällor som kan kopplas till differens mellan modell och verklighet på arbetsstycke, arbetsplats och robot.

Man får dessutom söka nya erfarenheter utifrån ny teknik. Speciellt gäller detta MIG/MAG-svetsning (Metal Inert Gas/Metal Active Gas) med robot, där man utnyttjar den erfarenhet som en svetsare har för att få ett fullgott resultat. De differenser som uppstår mellan modell/verklighet (arbetsstycke, arbetsplats och robot) ställer krav på bestämning av differenserna för att kunna kompensera för dessa fel.

Användningen av processororienterade språk innebär planering och lösning av komplexa uppgifter med tillgång till produkt- och processinformation. För dessa ändamål användes artificial intelligence (AI), även kallad machine intelligence eller heuristic programming. AI-tekniken utnyttjas i de problem där man inte analytiskt kan bestämma en bästa lösning på ett problem utan söker en lösning ur flera möjliga. Då verkliga problem oftast har ett stort antal beslutsmöjligheter, utnyttjar man en heuristisk sökteknik eller tumregler som talar om vilka beslut som är lämpliga eller olämpliga i en given situation. Områden som speciellt berör industriroboten är Natural Language Processing, Computer Vision, Expert Systems samt Problem Solving and Planning [3.5]. Det område som är kommersiellt tillgängligt för industrirobotar idag är visiontekniken, där man ger roboten syn för att bättre kunna arbeta mot en föränderlig omgivning. Användningen av dessa hjälpmedel är fortfarande mycket begränsad på grund av höga kostnader och låga prestanda.

3.1.6 SENSORER OCH SENSORSYSTEM

Med hjälp av sensorer kan industriroboten anpassa sitt arbete så att den kan korrigera sig för mindre förändringar som sker i omvärlden. I praktiken utnyttjas relativt enkla sensorer som ger binära signaler av on/off-typ. Uppbyggnaden av sensorarrangemangen görs dessutom ofta specialanpassade för varje ny produktserie. Förutom att utnyttja enkla binära givare finns det numera dessutom möjlighet att använda givare med ett utökat antal binära bitar eller som ger analoga signaler. En vanlig användning av sådana givare är sökning av föremål inom ett definierat område [3.6]. Vid mera avancerade sensorsystem som vision är det brukligt att sensorsystemet självt gör beräkningar som sedan ger korrektionsdata anpassade till styrsystemet.

Sensorer och sensorsystem kopplade till industrirobotar utnyttjas för i huvudsak två funktioner, bestämning av position och orientering samt kontroll av arbetsprocessen. Bestämning av position- och orientering bildar underlag för att ange hur robotens rörelsemönster skall korrigeras till överrensstämmelse med den verklighet som roboten arbetar i t ex plocka detaljer från en pall eller följa en svetsfog vid MIG/MAG-svetsning. Kontroll av arbetsprocessen kan inbegripa inspektion av detaljer och processövervakning.

Vid användandet av sensorer har utnyttjandet av sensorinformationen hittills varit begränsad till styrsystemets möjligheter att behandla informationen. Man kan också konstatera att industrimiljöer ställer mycket stora krav på sensorer och elektronik, varför det kommer att krävas stora utvecklingsinsatser för att nå en tillfredställande lösning. Exempel på störkällor är t ex ojämna ljusförhållanden och smuts på optik vid vision, magnetiska fält kring ljusbågen vid MIG/MAG-svetsning eller högfrekventa störningar från TIG-svetsning (Tungsten Inert Gas).

3.1.7 INTEGRERADE TILLVERKNINGSSYSTEM

För att få en svetsrobotstation flexibel och adaptiv vid småserier krävs en integrering av en mängd funktioner som t ex programmeringsmöjlighet on line och off line, hantering av processdata och sensoråterkoppling från sensorer i omgivningen samt materialhantering. Om man i en sådan station ekonomiskt kan svetsa en produktflora inom givna ramar (geometri, komplexitet, material, etc) i småserier, har man ett Flexible Manufacturing System (FMS) för svetsning. FMS-tekniken kan genom sammankoppling av flera tillverkningsceller med automatiska transportsystem möjliggöra produktion med begränsad bemanning eller helt obemannad drift. Flexibilitet är ett mångsidigt begrepp. I första hand avses här småserieproduktion och dess krav på en minimering av ställtider. I flexibilitet ligger också den beredskap som behövs för att i en befintlig produktionsutrustning snabbt kunna gå över till annan produktion eller kundorderstyrd tillverkning för att förbättra konkurrenssituationen. Flexibilitet kan också ligga i möjligheterna att variera kapacitetsutnyttjandet. Genom en ökad automation har man således skapat möjlighet att använda systemet under ett varierat kapacitetsutnyttjande.

Den traditionella bilden av FMS är en verkstad med maskiner som har möjlighet att arbeta obemannade med ett flertal olika detaljer under automatiskt byte av verktyg och arbetsstycken. Arbetsstyckena fästs på standardiserade paletter som förs runt i ett automatiskt transportsystem till de olika maskinerna. Transportsystemet kan vara uppbyggt med AGV (Automated Guided Vehicle) eller som rullbanor [3.7].

I ett längre perspektiv kan man se FMS som en del i ett vidare koncept, Computer Intergrated Manufacture (CIM), där man söker en integrering av alla ingående delar i produktkedjan inkluderande CAD, CAM, inköp, lagerhållning, försäljningsorder, tidsstyrning, etc. En sådan utveckling kommer givetvis att dramatiskt förbättra möjligheterna till kundorderstyrd tillverkning, minska kostnaderna för varor i arbete och lagerhållning, ge hög beredskap för flexibelt volymsutnyttjande av produktionsutrustningen (möjlighet till obemannad drift vid nattsift) samt reducera det manuella arbetet på verkstadsgolvet.

3.2 PRAKTISK ANVÄNDNING AV INDUSTRIROBOTEN

Vid robotsvetsning med MIG/MAG-metoden utnyttjas normalt en robot med 5-6 rörelseaxlar och en lägesställare som har möjlighet att förflytta objektet för att på så sätt få tillgänglighet och lämplig orientering av objektet relativt svetspistolen. MIG/MAG-metoden uppvisar många fördelar som är av stor betydelse vid robotsvetsning. Bland dessa kan nämnas kontinuerlig tillförsel av tillsatsmaterial, eliminering av slagg genom skyddsgas vid solidelektrod, svetsning i olika positioner, hög produktivitet samt möjlighet till kvalitetsstyrning. För att uppnå önskade resultat krävs stabila arbetsförhållanden under processen. Några faktorer som bestämmer svetsresultatet är preparering av detaljerna som skall svetsas (toleranser, ytfinnish), fixturkonstruktion, häftsvetsning, deformationer under svetsning på grund av värmepåverkan samt fel som kan härledas till roboten, exempelvis programfel eller toleransfel [3.8]. Vid korta seriestorlekar (50 enheter eller ett par timmars produktion) får man problem med lönsamheten p g a ständigt återkommande omprogrammering, omriggning och intrimning av svetsparametrar. Hjälpmedel börjar nu tas fram, där man med hjälp av fogsökare automatiskt kan söka sig fram till en

svetsfog [3.9] och speciella sensorsystem som har möjlighet att utvärdera svetsfogens läge och geometri och utifrån dessa data ge underlag för val av svetsparametrar [3.10] [3.11]. I praktiken har dock sensorbaserad robot-svetsning utvecklats långsamt av både tekniska och ekonomiska skäl. Dagens kommersiella styrsystem är inte anpassade att styras av sensorer i realtid, utan det normala är att man vid vissa på förhand bestämda tidpunkter mäter fogens position. Då produktiviteten redan är låg p g a ständigt återkommande omprogrammering, omriggning och inkörning av nya produkter blir det därför svårt att bära en ytterligare kostnad av ett sensorsystem som ligger i närheten av vad själva roboten kostar. Området är emellertid under stark utveckling och man kan förvänta sig en gynnsam pris/prestandautveckling i framtiden.

3.3 REFERENSER

- 3.1 B. M. Smith, et al, A Glossary of Terms for Robotics, PB82-251 216, National Bureau of Standards, Department of Commerce, Washington, D.C., Oct 1980.
- 3.2 H. J. Warnecke, R. D. Schraft, Industrial Robots: Application Experience, 1982, IFS(publ), UK.
- 3.3 Unimate Puma Robot, User's Guide to VAL, June 1980, Unimation inc.
- 3.4 L. I. Lieberman, M. A. Wesley, AUTOPASS: An Automatic Programming System for Computer Controlled Mechanical Assembly, IBM J. Res. Dep, July 1977.
- 3.5 W. B. Gevarter, An Overview of Artificial Intelligence and Robotics, Vol I, June 1983, NASA Technical Memorandum 85836.
- 3.6 ASEA Programming Manual IRb6/2, June 1984, ASEA Robotics inc.
- 3.7 J. Hartley, FMS at Work, IFS (publ) Ltd, UK, 1984.
- 3.8 L. G. Eriksson and G. Kullberg, The Development of Automatic Welding Systems for Small Lot Production, Automation and Robotisation in Welding and Allied Processes, Proc of the Int Conf, Strasbourg, France, 2-3 sept 1985.

- 3.9 G. Edling, T. Porsander, Adaptive Control of Torch Position and Welding Parameters in Robotic Arc Welding - Examples and Practice Use, Proc of the 14th Int Symp on Industrial Robots, Oct 1984, Gothenburg, Sweden.

- 3.10 R. Niepold, F. Brummer, PASS - A Visual Sensor for Seam Tracking and On Line Process Parameter Control in Arc Welding Applications, Proc of the 14th Int Symp on Industrial Robots, Oct 1984, Gothenburg, Sweden.

- 3.11 W. J. P. A. Verbeck, Arc Welding Process Control by Preview Sensor, pp 86-88, The Industrial Robot, June 1984.

4. UTVECKLING AV ROBOTSTATIONER FÖR BÅGSVETSNING

Den kommersiella utvecklingen av industrirobotsystem har lett fram till en betydande användning av robotar för bågsvetsning. Anledningen till detta skall ses mot bakgrund av en rad faktorer som tillsammans påskyndar användningen och utvecklingen mot en mera flexibel och automatiserad svetsning. Exempel på sådan faktor är bristen på kvalificerade svetsare. Detta gäller både där kvalitetskraven är höga och i tempobetonad serieproduktion och där arbetsmiljön medför svårigheter att rekrytera arbetskraft. En annan faktor är att produktiviteten vid robotsvetsning ökar avsevärt i och med att bågtdidsfaktorn som vid manuell svetsning är ca 25% kan nå upp till 80% vid robotsvetsning. Den främsta anledningen till detta är en snabb förflyttning mellan svetsarna samt att roboten inte har ett behov att återhämta sig vid krävande och ansträngade arbetsuppgifter. Detta indikerar att man kan förvänta sig en högre produktivitet vid många korta svetsar än vid detaljer som har ett fåtal långa svetsar. Under väl definierade betingelser kan man räkna med en hög och framför allt jämn kvalitet. Detta medför en enklare och säkrare efterkontroll av produkterna liksom ett minskat behov av efterbearbetning och justering.

I vissa fall kan man med robot utföra svetsning som är olämplig eller som inte kan utföras manuellt på grund av miljöskäl eller begränsad åtkomst. Bland miljöskälen kan nämnas rökgaser och stoft, buller, bländning, ultraviolett strålning, hög temperatur samt fysisk belastning.

Beträffande flexibilitet är den manuella svetsaren överlägsen varje annan mekaniserad svetsning. Industrirobotar för bågsvetsning kan dock användas vid en begränsat varierad produktflora i lämpliga seriestorlekar. Industriroboten i sig betingar dessutom ett restvärde som kan utnyttjas i annan produktion, t ex hanteringsuppgifter eller maskinbetjäning.

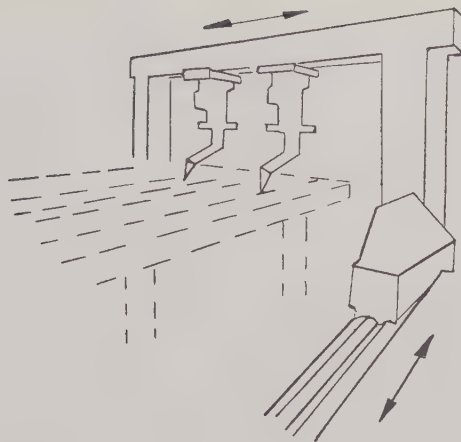


Bild 4.1. Exempel på robotar monterade på åkbana för svetsning av större arbetsstycken.

Den centrala delen av bågsvetsstationen har hittills varit industriroboten vilken utnyttjats för att utföra själva svetsoperationen, dvs föra en svetspistol med korrekt orientering och hastighet längs en fog. Vid större arbetsobjekt räcker inte robotens arbetsområde till. En vanlig teknik är då att hänga roboten upp och ned och göra den flyttbar, se bild 4.1.

Vid denna typ av svetsning utnyttjas ett åkdonsystem som för roboten till svetsfogens arbetsområde så att roboten kan upprätthålla korrekta pistolvinklar vid svetslägena. I och med detta kommer man redan vid projekteringen av en robotsvetsstation att sätta gränserna för flexibiliteten, åtminstone vad gäller den produktmix som kan svetsas i stationen.

Vid mindre objekt, där man har en kortare svetstid och där seriestorlekarna minskar mot medelstora serier eller småserier, ställs mycket höga krav på robotens möjlighet att svetsa olika typer av produkter (fogtyper, foglägen). En vanlig lösning på problemet är då (om detaljerna är av hanterlig storlek och vikt) att låta fästa fixtur och objekt på en lägesställare, se bild 4.2. Avsikten med detta arrangemang är att lägesställaren skall kunna placera

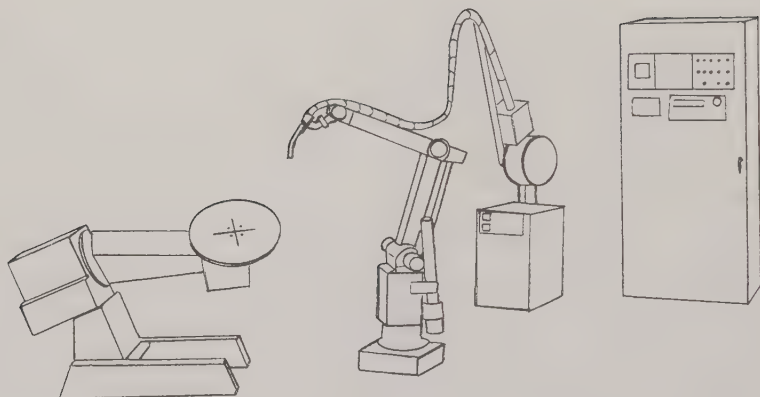


Bild 4.2. Svetsrobotstation med lägesställare för svetsning av mindre objekt.

objektet så att man erhåller en lämplig orientering av svetsfogen och/eller så att svetsfogen kommer inom robotens arbetsområde samt att svetspistolen kan orienteras relativt fogen på lämpligt sätt. Ett annat sätt att utnyttja lägesställaren är att samtidigt röra både robot och lägesställare så att korrekt fogläge eller åtkomlighet vid långa eller kurvformiga fogar uppnås. Ett problem som härvid uppträder är att åstadkomma en kinematisk beskrivning av robot och lägesställarrörelserna som medger en synkron rörelse av bägge enheterna. Genom detta kan robotens TCP styras utefter ett rörligt objekt som är fast kopplat till lägesställaren.

Vid robotsvetsning är det vanligt förekommande att samma styrsystem som styr roboten också styr svetsparametrar som ström och spänning. Detta ger ett enklare handhavande vid programmering och inkörning. Efter inkörning får man härmed automatiskt de injusterade värdena lagrade i eller tillsammans med programmet för återkommande tillverkning. I och med att svetsprocessen skall kunna fortgå automatiskt och utan mänsklig kontroll ställs speciella krav på strömkällor för robotsvetsning. Bland dessa kan nämnas tillförlitlighet,

styrmöjligheter och ljusbågförhållande. Speciellt intressant under senare år är utvecklingen av transistoriserade högfrekvent pulsan­de strömkällor med mycket goda styrmöjligheter och möjlighet till sprutfri svetsbåge.

I de följande avsnitten kommer jag att ta upp den nuvarande statusen hos ingående delar i robotstationer för bågsvetsning vid småserieproduktion och hur dagens teknik utnyttjas i kompletta system.

4.1 ROBOTSYS­TEM FÖR BAGSVETSNING

Robotsystemet som arbetar i en robotsvetsstation består av ett flertal samverkande delar, se bild 4.3. De funktionskrav som ställs vid bågsvetsning är

- huvudmanipulator (robot) och hjälpmanipulator (lägesställare) som har till uppgift att upprätthålla ett korrekt samspel så att svetspistol och objekt får för processen lämpliga positioner och orienteringar
- styrsystem med uppgift att understödja programmeringsarbetet, kunna kommunicera med externa system och koordinera styrningen av svetsmaskin (svetsparametrar), robot och lägesställare vid svetsoperationer
- externa sensorer som har till uppgift att bestämma och kontrollera tillstånd som är av betydelse för svetsprocessen

4.1.1 MEKANISK STRUKTUR

Vid bågsvetsning krävs en mekanisk struktur som ger god åtkomlighet för att positionera och orientera svetspistolen mot objektet. För detta ändamål har de revoluta robottyperna kommit att helt dominera vid bågsvetsning. Ett problem vid svetsning med robotar är de krav på definierad banstyrning (linjärt, cirkulärt) och hastighet som processen ställer. Utöver detta ställs krav eller önskemål att kunna röra objektet under svetsning i ett optimalt fogläge och med fogen inom robotens arbetsområde. Vi skall nu först studera de metoder som används för att skapa en modell över kinematiken varigenom CP-styrning möjliggöres.

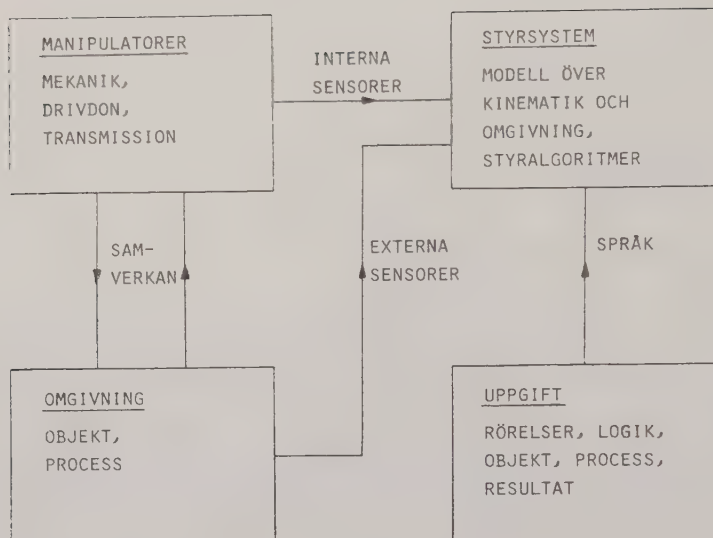


Bild 4.3. Schematisk bild av samverkande delar i ett robotsystem för robotsvetsning.

Vid bågsvetsning ställs krav på CP-styrning som ger en definierad rörelse till position, orientering och hastighet. Problemet är hur man skall beskriva kinematiken för att erhålla en datorbaserad CP-styrning. Detta kan delas upp i två delar, det direkta kinematiska problemet och det inversa kinematiska problemet. Det direkta kinematiska problemet är att finna en beskrivning som ger verktygets eller svetspistolens position och orientering givet robotarmens ledvinklar (och slidpositioner om det ingår translationsrörelser i systemet), medan det inversa kinematiska problemet är att finna robotens ledvinklar (eller slidpositioner) givet verktygets position och orientering, se bild 4.4. För 5- eller 6-axliga robotar kommer överföringsfunktionerna mellan det robotorienterade koordinatsystemet (ett koordinatsystem som bestäms utifrån robotarmens leder) och ett kartesiskt baskoordinatsystem att bli av en sådan svårighetsgrad att man idag måste ställa stora krav på snabbhet i hårdvara och mjukvara för att styrrutinerna skall kunna användas i realtid.

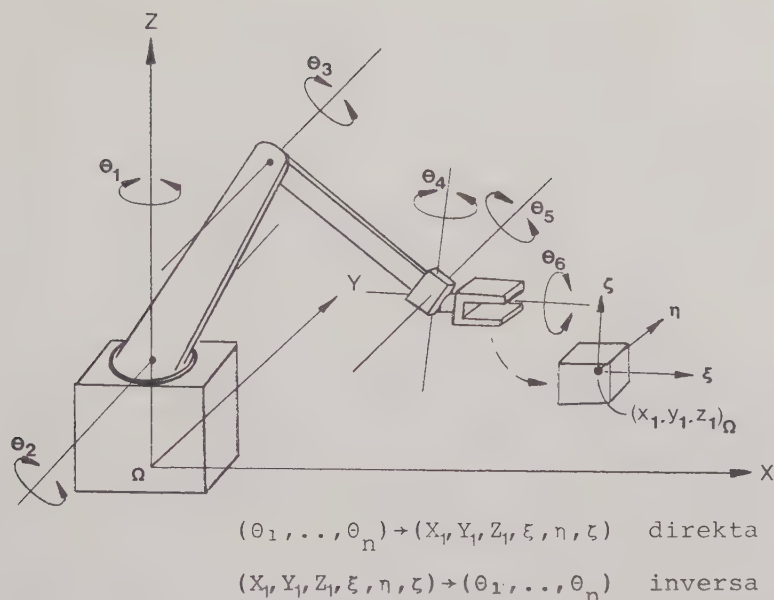


Bild 4.4. Det direkta och inversa kinematiska problemet för en robotarm.

Vid praktiskt taget alla kinematiska modeller över robotar betraktas robotarmen som en serie av stela kroppar förbundna med varandra i rotations- eller translationsleder. För den revoluta robottypen gäller att samtliga leder är av rotationstyp. Vid användning av en kinematisk modell för styrning av robotens rörelser måste man komma ihåg att man inte har tagit hänsyn till hur tröghetskrafter, massor, toleranser och transmissioner påverkar robotens uppträdande. Varför dessa begränsningar görs och hur de påverkar robotens uppträdande vid bågsvetsning kommer att belysas senare i detta kapitel. Vi skall nu närmare studera vilka metoder som används för att skapa en kinematisk modell över en robotarm och vilka problem som uppstår med de olika metoderna.

Om varje ledpar antas innehålla en frihetsgrad (rotation eller translation), finns det n ledpar med led 0 i baskoordinatsystemet på en robotarm med n frihetsgrader eller rörelseaxlar. För att beskriva relationerna mellan två länkar är en vanlig metod den som beskrivs av Denavit - Hartenberg's [4.1] enligt följande:

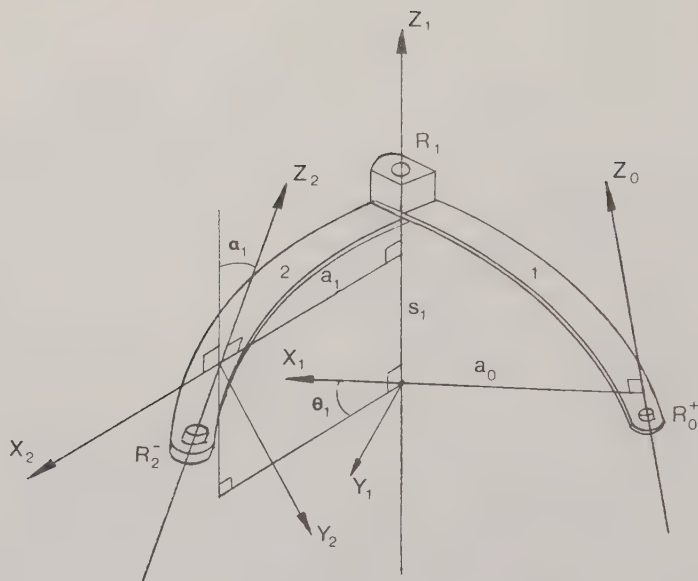


Bild 4.5. Koordinatrepresentation för ett revolut länkpar enligt Denavit Hartenberg [4.1].

Betrakta ett revolut länkpar R_1 enligt bild 4.5 med armarna 1 och 2 och med angränsande leder R_0^+ och R_2^- . Axlarna z_0 , z_1 och z_2 definierar länkmekanismens ledpositioner utifrån ett baskoordinatsystem. Axeln x_1 väljs som en förlängning av det vinkelräta avståndet a_0 mellan axlarna z_0 och z_1 . Axeln y_1 väljs så att man får ett högerorienterat koordinatsystem. s_1 är det vinkelräta avståndet mellan x_1 och x_2 , α_1 är vridningsvinkeln mellan z_1 och z_2 -axlarna och θ_1 är vridningsvinkeln mellan x_1 och x_2 -axlarna. På detta sätt kommer leden R_2^- att vara helt definierad i förhållande till R_1 genom de fyra parametrarna a_1 , θ_1 , s_1 och α_1 .

Metoden kan generaliseras för att användas på robotarmar som består av successivt förbundna länkarmar med baskoordinatsystemet som system 0, den första leden som system 1 och så vidare ut till den i :te leden. Vid transformering av koordinatsystem används homogena transformationsmatriser som beskriver koordinatsystemets position och orientering i rummet. En homogen transformationsmatris är en 4×4 matris som överför ett objekt

definierat i ett homogent koordinatsystem. En homogen transformationsmatrix kan därför i detta sammanhang ses som två submatriser, en rotationsmatrix och en translationsmatrix:

$$T = \left[\begin{array}{c|c} \text{rotations-} & \text{translations-} \\ \text{matrix} & \text{matrix} \\ 4 \times 3 & 4 \times 1 \end{array} \right]_{4 \times 4} \quad (4-1)$$

När koordinatsystemen har definierats enligt ovan kan en homogen transformationsmatrix A_{i-1} skapas för varje länkpar som ger position och orientering för led_i relativt led_{i-1}. För en revolut led får A_{i-1} följande utseende:

$$A_{i-1}^i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & s_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4-2)$$

En transformationsmatrix för hela robotarmen kan sedan skapas genom matrismultiplikation som ger position och orientering av det i:te koordinatsystemet relativt baskoordinatsystemet enligt följande:

$$T_0^i = \prod_{j=1}^i A_{j-1}^j = \begin{bmatrix} x_i & y_i & z_i & p_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4-3)$$

där x_i , y_i och z_i ger det ite koordintsystemets orientering och p_i dess position.

Metoden är vanligt förekommande och ger en rättfram lösning på det direkta kinematiska problemet [4.2] [4.3] [4.4] [4.5]. För en 6-axlig PUMA-robot får man med metoden en lösning av koordinattransformationsmatriserna enligt bild 4.6.

Ur bild 4.6 ges lösningen på armmatrisen som

$$T = A_0^1 A_1^2 A_2^3 A_3^4 A_4^5 A_5^6 \quad (4-4)$$

och man erhåller följande lösning för verktygets orientering och position:

$$n_x = C_1 [C_{23}(C_4 C_5 C_6) - S_{23} S_5 C_6] - S_1 [S_4 C_5 C_6 + C_4 S_6] \quad (4-5)$$

$$n_y = S_1 [C_{23}(C_4 C_5 C_6 - S_4 S_6) - S_{23} S_5 C_6] + C_1 [S_4 C_5 C_6 + C_4 S_6] \quad (4-6)$$

$$n_z = -S_{23} [C_4 C_5 C_6 - S_4 S_6] - C_{23} S_5 C_6 \quad (4-7)$$

$$s_x = C_1 [-C_{23}(C_4 C_5 S_6 + S_4 C_6) + S_{23} S_5 S_6] - S_1 [-S_4 C_5 S_6 + C_4 C_6] \quad (4-8)$$

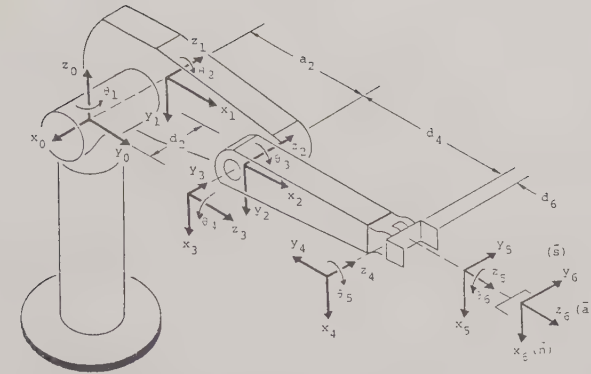
$$s_y = S_1 [-C_{23}(C_4 C_5 S_6 + S_4 C_6) + S_{23} S_5 S_6] + C_1 [-S_4 C_5 S_6 + C_4 C_6] \quad (4-9)$$

$$s_z = S_{23}(C_4 C_5 S_6 + S_4 C_6) + C_{23} S_5 S_6 \quad (4-10)$$

$$a_x = C_1 (C_{23} C_4 S_5 + S_{23} C_5) - S_1 S_4 S_5 \quad (4-11)$$

$$a_y = S_1 (C_{23} C_4 S_5 + S_{23} C_5) + C_1 S_4 S_5 \quad (4-12)$$

$$a_z = -S_{23} C_4 S_5 + C_{23} C_5 \quad (4-13)$$



PUMA armparametrar					
Joint i	θ_i	α_i	a_i	d_i	
1	90	-90	0	0	
2	0	0	432	149.5	
3	90	90	0	0	
4	0	-90	0	432	
5	0	90	0	0	
6	0	0	0	56.5	

$$C\theta_i = \cos\theta_i ; S\theta_i = \sin\theta_i$$

$$A_0^1 = \begin{bmatrix} C\theta_1 & 0 & -S\theta_1 & 0 \\ S\theta_1 & 0 & C\theta_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1^2 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & a_2 C\theta_2 \\ S\theta_2 & C\theta_2 & 0 & a_2 S\theta_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^3 = \begin{bmatrix} C\theta_3 & 0 & S\theta_3 & 0 \\ S\theta_3 & 0 & -C\theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^4 = \begin{bmatrix} C\theta_4 & 0 & -S\theta_4 & 0 \\ S\theta_4 & 0 & C\theta_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4^5 = \begin{bmatrix} C\theta_5 & 0 & S\theta_5 & 0 \\ S\theta_5 & 0 & -C\theta_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_5^6 = \begin{bmatrix} C\theta_6 & -S\theta_6 & 0 & 0 \\ S\theta_6 & C\theta_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Bild 4.6. Lösning av det direkta kinematiska problemet för en PUMA-robot enligt Denavit - Hartenberg's metod [4.3].

$$p_x = C_1 [d_6 (C_{23} C_4 S_5 + S_{23} C_5) + S_{23} d_4 + a_2 C_2] - S_1 (d_6 S_4 S_5 + d_2) \quad (4-14)$$

$$p_y = S_1 [d_6 (C_{23} C_4 S_5 + S_{23} C_5) + S_{23} d_4 + a_2 C_2] + C_1 (d_6 S_4 S_5 + d_2) \quad (4-15)$$

$$p_z = d_6 (C_{23} C_5 - S_{23} C_4 S_5) + C_{23} d_4 - a_2 S_2 \quad (4-16)$$

där n är normalvektorn mot verktyget (motsvarar x_6), s är 'sliding' vektor för verktyget (motsvarar y_6), a är approach vektor för verktyget (motsvarar z_6), p är positionsvektorn för verktyget, $C_i = \cos \theta_i$, $S = \sin \theta_i$ och $S_{ij} = \sin(\theta_i + \theta_j)$.

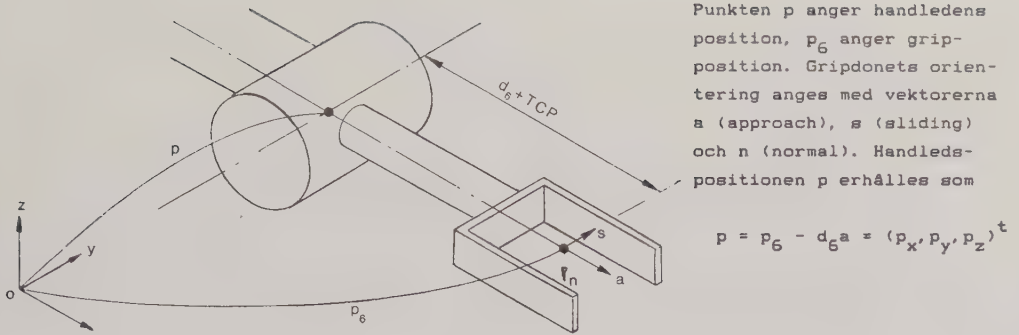
Som framgår av lösningen innehåller den ett stort antal tidskrävande operationer som består av flyttal och trigonometriska funktioner. Vid lösandet av det direkta kinematiska problemet erhålls alltid en lösning. Detta är inte alltid fallet för det inversa kinematiska problemet som vi skall se nedan.

För att en explicit lösning skall kunna erhållas vid det inversa kinematiska problemet ställs krav på hur robotens kinematiska struktur ser ut. I fallet med en revolut robot krävs sex frihetsgrader, där dessa kan delas upp så att tre frihetsgrader utför positioneringen och tre frihetsgrader utför orienteringen av handen eller verktyget. Dessutom krävs det att de tre handledsaxlarna är skärande. Exempel på en lösning av det inversa kinematiska problemet med ett geometriskt betraktelsesätt presenteras i bild 4.7 enligt [4.3].

Trots att det finns en explicit lösning som ger roboten ledvinklar om man ger en position och orientering för verktyget, kan man inte garantera en lösning beroende på begränsningar i armlängder och vridningsvinklar hos robotarmen. I vissa fall erhålles dessutom singulariteter i lösningen. På grund av detta krävs beslutsfunktioner som bestämmer hur lösningen skall se ut (om det finns någon) beroende på kinematikens eller den numeriska lösningens utseende.

För en 5-axlig robot kan en explicit lösning endast tas fram för positionen i det inversa fallet eftersom man inte kan räkna med att kunna orientera verktyget på ett önskat sätt med endast fem frihetsgrader (den orientering som en 5-axlig robot klarar uppnås i singulariteten för θ_4 i bild 4.7). Handledens rörelser får istället te sig anges direkt i handledskoordinater och interpoleras linjärt över dessa under en rörelse.

Som man ser av bild 4.7 och resonemanget ovan är lösningen av det inversa problemet av icke trivial natur, speciellt om man betänker att beräkningarna skall göras i realtid om intervall på 10 - 30 ms. Av denna anledning läggs mycket arbete ner på att söka optimala algoritmer för specifika robotarmar



$$\theta_1 = \arctan \left[\frac{\pm p_y \sqrt{p_x^2 + p_y^2 - d_2^2} - d_2 p_x}{\pm p_x \sqrt{p_x^2 + p_y^2 - d_2^2} + d_2 p_y} \right]$$

$$\theta_3 = \arctan \left[\frac{p_x^2 + p_y^2 + p_z^2 - d_4^2 - a_2^2 + d_2^2}{\pm \sqrt{4d_4^2 a_2^2 - (p_x^2 + p_y^2 + p_z^2 - d_4^2 - a_2^2 - d_2^2)^2}} \right]$$

$$\theta_2 = \arctan \left[\frac{-[p_z(a_2 + d_4 S \theta_3) + (d_4 C \theta_3)(\pm \sqrt{p_x^2 + p_y^2 - d_2^2})]}{p_z(d_4 C \theta_3) - (a_2 + d_4 S \theta_3)(\pm \sqrt{p_x^2 + p_y^2 - d_2^2})} \right]$$

$$\theta_4 = \arctan \left[\frac{C \theta_1 a_y - S \theta_1 a_x}{C \theta_1 C \theta_{23} a_x - S \theta_1 C \theta_{23} a_y - S \theta_{23} a_z} \right]$$

$$\theta_5 = \arctan \left[\frac{(C \theta_1 C \theta_{23} C \theta_4 - S \theta_1 S \theta_4) a_x + (S \theta_1 C \theta_{23} C \theta_4 + C \theta_1 S \theta_4) a_y - C \theta_4 S \theta_{23} a_z}{C \theta_1 S \theta_{23} a_x + S \theta_1 S \theta_{23} a_y + C \theta_{23} a_z} \right]$$

$$\theta_6 = \arctan \left[\frac{(-S \theta_1 C \theta_4 - C \theta_1 C \theta_{23} S \theta_4) n_x + (C \theta_1 C \theta_4 - S \theta_1 C \theta_{23} S \theta_4) n_y + (S \theta_4 S \theta_{23}) n_z}{(-S \theta_1 C \theta_4 - C \theta_1 C \theta_{23} S \theta_4) s_x + (C \theta_1 C \theta_4 - S \theta_1 C \theta_{23} S \theta_4) s_y + (S \theta_4 S \theta_{23}) s_z} \right]$$

S anger sinus, C anger cosinus, θ_{23} anger $\theta_2 + \theta_3$; $-\pi \leq \theta_1 \leq \pi$

$\arctan$ definieras som $\theta = \arctan \begin{cases} 0^\circ \leq \theta \leq 90^\circ & \text{för } +x, +y \\ 90^\circ \leq \theta \leq 180^\circ & \text{för } -x, +y \\ -180^\circ \leq \theta \leq -90^\circ & \text{för } -x, -y \\ -90^\circ \leq \theta \leq 0^\circ & \text{för } +x, -y \end{cases}$

Bild 4.7. Lösning av det inversa kinematiska problemet för en 6-axlig PUMA robot. Armparametrarna ges ur bild 4.6.

för att på detta sätt minska beräkningstiderna i styrdatorn [4.5] [4.6]. Om det inte är möjligt att ta fram en explicit lösning kan iterativa algoritmer användas enligt [4.7]. En metod bygger på antagandet att de tre första rörelseaxlarna i huvudsak påverkar verktygets position och att de tre sista axlarna i huvudsak påverkar verktygets orientering. Under denna förutsättning kan man dela upp robotarmen i två grupper, positionerings- och orienteringsaxlar. Position och orientering beräknas då successivt var för sig tills

felet har konvergerat till ett tillåtet värde. Den iterativa processen kan då beskrivas enligt följande:

1. beräkna positionsaxlarnas vinklar utifrån önskad position och med konstanta värden för orienteringsaxlarna $(\theta_o) = \theta_p$
2. beräkna verktygets orientering utifrån beräknade positionsvinklar i (1) och handledsaxlar $(\theta_o) = X_o$
3. om orienteringen X_o ligger inom feltoleransen går vi ur processen
4. beräkna orienteringsaxlarnas vinklar utifrån (1) och (2), $= \theta'_o$
5. beräkna aktuell position utifrån (1) och (4), $= X'_p$
6. om aktuell position ligger inom feltoleransen går vi ur processen
7. gå till (1)

Bland fördelarna av den beskrivna metoden märks att den möjliggör eller förenklar sökningen av en lösning för en robotarm. Konvergensen mot en lösning i processen är dock starkt beroende av förhållandet mellan längderna på de tre första armarna och de tre sista samt placeringen av TCP i robotens arbetsvolym. För att minska konvergensproblemet kan man söka en linjärisering av den direkta kinematiska transformationsmatrisen för att erhålla ett linjärt ekvationssystem som approximativt bestämmer hur vinklarna skall vara kring en arbetspunkt. Eftersom man vid ett interpoleringsförfarande beräknar nya arbetspunkter med hög täthet, kommer startvärdena i iterationsprocessen i detta fall inte att vara något problem.

Man kan mot bakgrund av de beskrivna metoderna för transformering mellan de olika koordinatsystemen konstatera vilka svårigheter som uppstår om man vill utöka modellen av roboten med hur tröghetskrafter, massor, toleranser och transmissioner påverkar robotens arbetssätt. Tröghetskrafter och elasticitet i robotarmen kommer att påverka systemet i större utsträckning vid högre hastigheter. För bågsvetsning gäller emellertid att själva processen normalt utförs vid låga hastigheter (ca 10 mm/s). Vid positionering mellan svetsar däremot ställs höga krav på hög acceleration och retardation. Vid denna positionering har man emellertid inga speciella krav på hur rörelsen utförs mellan

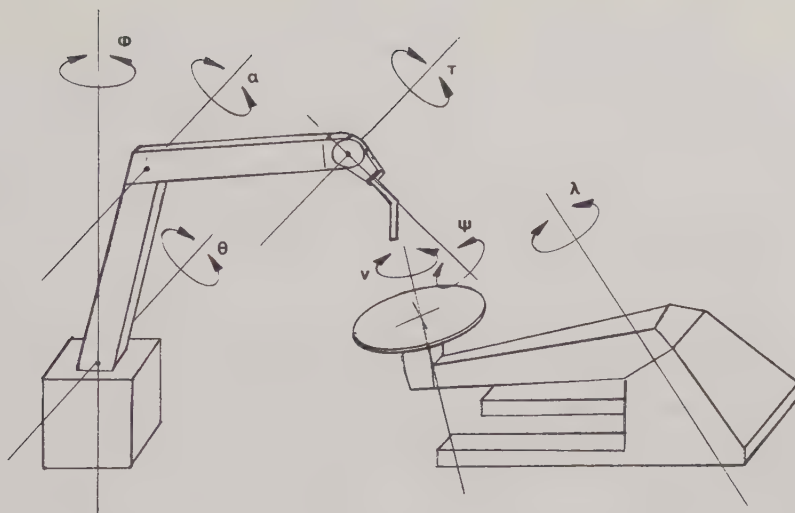


Bild 4.8. Robotstation med sju rörelseaxlar fördelat på fem axlar för robot (svetspistol) och två axlar för lägesställare (objekt).

två positioner. Man kan därför säga att den kinematiska modellen ger en tillfredställande beskrivning av robotarmen för en datorbaserad styrning av rörelserna. Av speciellt intresse vid bågsvetsning är fallet, där en lägesställare ingår i stationen och svetsning utförs mot ett rörligt objekt. Med en 5-axlig robot och en 2-axlig lägesställare innebär detta att det kommer att finnas totalt sju rörelseaxlar i systemet, se bild 4.8.

För lägesställaren gäller det primärt att orientera aktuell fog så att man erhåller en orientering som ger bästa svetsresultat. Om det finns flera möjliga orienteringar eller om fogen ligger helt eller delvis utanför robotens arbetsområde, krävs beslutsrutiner som bestämmer hur fogen skall orienteras så att roboten kan orientera svetspistolen på ett lämpligt sätt för att erhålla ett gott svetsresultat. Av detta följer att även om den kinematiska styrmodellen kan uttryckas explicit, krävs det beslutsfunktioner som är processrelaterade för att man skall erhålla ett tillfredställande resultat vid generering av manipulatorrörelserna.

4.1.2 STYRSYSTEM

Styrssystemet är robotstationens centrala del som utifrån en definierad uppgift styr och kontrollerar det fysiska händelseförloppet i stationen. Funktionskrav som är starkt knutna till robotsvetsning gäller de tidigare nämnda problemen som rör programmering, CP-styrning, anpassning av processen efter sensorer och då speciellt för att kunna återkoppla rörelsen genom fogföljning. Vi skall nu studera hur dessa tekniker används i dagens system.

Ett grundkrav för att ett styrsystem skall kunna användas av en svetsrobot är att styrsystemet kan generera en rörelse som är definierad till position och hastighet, ofta genom linjära eller cirkulära rörelser. För en linjär rörelse av TCP kan beräkningsgången då bli följande, se bild 4.9:

$$X(t) \rightarrow \Theta(t) \rightarrow I(t) \rightarrow V(t) \rightarrow \text{fysisk rörelse av robotaxlarna}$$

där $X(t)$ beskriver rörelsen i ett kartesiskt koordinatsystem över tiden, $\Theta(t)$ beskriver rörelsen i robotkoordinater (ledvinklar), $I(t)$ beskriver rörelsen utifrån drivdonen (inkrementella rörelser på drivdonen som krävs för att vrida robotarmarna $\Theta(t)$ radianer) och slutligen $V(t)$ som beskriver hur drivmotorerna skall drivas ut för att ge önskad rörelse på robotaxlarna.

Om samplingsfrekvensen i styrsystemet är 100Hz, innebär detta att nya positionsdata måste beräknas var 10:e ms. För en linjär interpolation över en sträcka på s mm och med hastigheten v mm/s innebär detta att rörelsens positioner över tiden kommer att beräknas $s/v * f$ gånger (f är frekvensen eller 100Hz i detta fallet).

Vid positionsstyrning förutsättes enligt de antaganden som gjordes i föregående avsnitt att man för bågsvetsning kan använda sig av en kinematisk modell som är beskriven tidigare. Den för bågsvetsning intressanta delen är hur styrsystemet fås att röra sig utefter en definierad bana och med definierad hastighet, dvs i huvudsak interpolering i kartesiska koordinater och överföringsfunktionen från kartesiska koordinater till robotkoordinater över

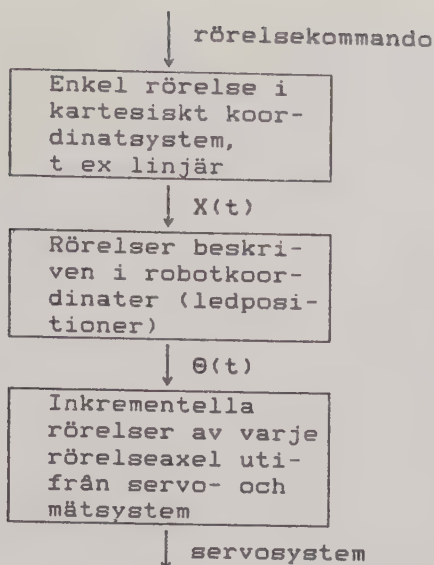


Bild 4.9 Beräkningar som utförs vid en CP-styrd rörelse.

tiden ($X(t) \rightarrow \Theta(t)$). En interpolering över enkla kurvor i ett kartesiskt koordinatsystem som linjära och cirkulära rörelser är ett trivialt problem och erbjuds också på kommersiella system [4.8]. Problemet är snarare hur överföringen $X(t) \rightarrow \Theta(t)$ skall gå till utan att styrsystemet råkar i tidsnöd och att man samtidigt har möjlighet att i realtid korrigera rörelsen med hänsyn till fögföljningsproblematiken.

Ett sätt att lösa problemet vid tidsnöd är att sänka samplingsfrekvensen för att ge styrdatorn mer tid att räkna ut nya robotkoordinater och sedan göra en interpolering i robotkoordinater mellan positionspunkterna, se bild 4.10.

En faktor som komplicerar bilden vid bågsvetsning är att man ofta använder sig av en lägesställare för att manipulera objektet. Om man av processtekniska skäl önskar röra både robot och lägesställare samtidigt under svetsning, krävs det att styrsystemet har en kinematisk modell över både robot och lägesställare som gör det möjligt att beskriva och utföra en CP-styrning av svetspistolen utefter en rörlig svetsfog. Om motsvarande styrning skall

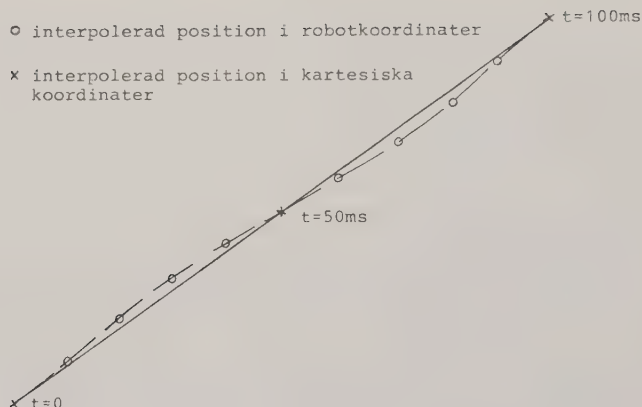


Bild 4.10 Modifierad interpoleringsteknik för att klara tidskravet vid CP-styrning.

utföras på dagens system, måste rörelsen programmeras in som på PTP-system utan TCP-funktion.

På grund av svårigheterna som är förknippade med CP-styrningen enligt ovan finns det praktiskt taget inga robotsystem idag som kan arbeta med fogföljningstekniken i praktisk verkstadsmiljö. I ett kommande avsnitt i detta kapitel skall jag behandla sensorsystem som utvecklas i syfte att användas för fogföljning.

En viktig del av dagens styrsystem är utnyttjandegraden vid programmeringen av roboten. Den vanligen använda tekniken idag är on line programmering, där man via en programmeringslåda med joystick och menyer ger kommandon och styr roboten till de positioner man önskar definiera. Vid ofta återkommande programmering och program med många logiska instruktioner uppstår den nackdelen att robotsystemet kommer att vara upptaget för programmering en relativt hög del av sin totala tid jämfört med produktiv tid. Situationen idag är den att en större del av programmeringsinstruktionerna är av rörelsetyp, men genom införandet av FMS och obemannad drift, där sensorer får en allt större betydelse, kommer de logiska instruktionerna i programmen att öka i antal jämfört med rörelseinstruktioner.

För att reducera programmeringstiden vid roboten strävar man mot off line programmering, dvs att i huvudsak utnyttja ett annat programmeringssystem för att generera instruktioner till roboten. Idag används off line programmering i praktiken i ytterst liten utsträckning och då den används utnyttjar man som regel endast robotorienterade programspråk. En anledning till detta är att off line programmeringen leder till en helt ny arbetsmetodik ur vilken erfarenheter måste hämtas för att metoden skall bli konkurrenskraftig. En annan anledning är att många programspråk fortfarande är under stark utveckling och att det saknas standard på området. En av de avgörande faktorerna är emellertid de noggrannhetsproblem som uppträder då robotens rörelse räknas fram i stället för att läras in som vid on line programmering. De fel som kommer in i beräkningarna härrör huvudsakligen från toleranser i robotarmens dimensioner som medför en inkorrekt kinematisk beskrivning, toleransfel i uppmätning av TCP, fel i den modell mot vilken roboten skall arbeta samt mekaniska och termiska spänningar i robotarmen. För att komma tillrätta med detta utvecklas korrigeringsalgoritmer som kring ett arbetsområde translaterar och/eller roterar aktuella rörelser enligt det uppmätta felet [4.9].

Trots den stora floran av programspråk kan man urskilja en del gemensamma drag, där uppbyggnaden följer i huvudsak två filosofier. Den första är det i NC-sammanhang beprövade APT-språket (Automatically Programmed Tool), där språket har utökats med en robotspecifik kommandorepertoar. Den andra är ett traditionellt högnivåspråk för programmering, vanligen PASCAL eller BASIC, där kommandorepertoaren är utökad med robotspecifika kommandon [4.10] [4.11] [4.12]. Utöver dessa finns det en mängd programspråk definierade utan någon speciell koppling till existerande språk. Nackdelen med detta är att det försvårar användningen av olika robotar med var sitt programspråk genom de problem och risker för misstag som uppstår genom en sammanblandning av kommandon eller programmeringsprocedurer. Bland funktioner i programspråken som kan vara av speciellt intresse är möjligheten att definiera koordinerade eller simultana händelser mellan flera samverkande enheter (robot - omgivning eller robot - robot). För realtidstillämpningar uppvisar programspråket ADA stora potentiella möjligheter att genom sitt task-begrepp ge en klar uppdelning av ett program i processer som kan exekveras i parallella processorer [4.13]. Tekniken är emellertid för ny för att man skall kunna uttala sig om dess praktiska användbarhet i robotsammanhang.

Då roboten arbetar mot en omgivning sker en ömsesidig påverkan mellan robot och omgivning. Vid flexibel automation eller där slumpmässiga förändringar uppstår i omgivningen krävs det att robotsystemet har tillgång till sensor-information för att kunna modifiera eller anpassa sitt arbetssätt utifrån den verklighet som råder.

De krav som ställs på ett sensorinteraktivt styrsystem kan i stor utsträckning jämföras med de motoriska system som återfinns i naturen, där rörelser som genereras oftast är i harmoni med den omgivning som råder, vilket även gäller om omgivningen är föränderlig. Vid uppläggnen av strukturen för ett sensorinteraktivt styrsystem med egenskaper enligt diskussionen ovan är det därför av intresse att göra en jämförande studie av hur motsvarande styrmekanismer kan se ut i biologiska system.

Man kan konstatera att verksamheter som planering, beslutsval och problemlösning är fundamentala komponenter i intelligent beteende hos människa, djur eller maskiner. En definition av intelligens i detta sammanhang anges enligt [4.14] att vara

"...able to contemplate the potential costs and benefits of future actions, to imagine goals, and to plan sequences of actions directed towards the attainment of those goals may in fact be taken (at least in a part)...".

Man kan också notera att det är dessa uppgifter (problemlösning och planering) som man till stor del söker lösa inom AI [4.15], [4.16]. Teknikerna inbegriper vanligen binära sökträd med exempelvis heuristisk eller trial-and-error sökning av olika möjliga lösningar på det aktuella problemet. Ett grundläggande angreppssätt vid lösandet av komplexa problem är en uppdelning av problemet till delproblem som i sin tur delas upp i delproblem osv till dess att problemet har blivit nedbrutet i så små beståndsdelar att dessa kan lösas genom enstegslösningar (jmf högnivå- och assembler-programmering av datorer, där ett kommando i ett högnivåspråk motsvarar en komplex uppgift som bryts ner till assemblerkommandon som motsvarar

enstegslösningen på varje minsta delproblem). Avsikten med denna teknik är att delproblemen tillsammans skall lösa det totala problemet samt att de nedbrutna delproblemen var och en skall ge enkla lösningar. Användningen av AI-tekniken har hittills mest behandlat en total optimering av en lösning på ett problem, exempelvis spel som schack etc, där tiden har relativt liten betydelse. Denna metodik är dock inte direkt användbar på industrirobotar som arbetar i en process, där arbetssituationen kan ändras genom okontrollerade händelser i omgivningen eller processen. Styrsystemet kommer då inte att ha tid att planera eller lägga upp strategier för processen, utan det måste i första hand handla och agera i realtid, då en arbetsprocess i industrimiljö är snabbt föränderlig med slumpmässiga händelser som uppstår i tiden. Detta gör att ett planerande styrsystem, där ett tillstånd resulterar i en mängd aktiviteter som sedan utförs 'blint' (s k open loop) ofta tenderar att uppträda ointelligent, då oväntade händelser uppträder under de 'blinda' aktiviteterna.

Man kan konstatera att det för ett produktionsstyrssystem oftast är omöjligt att inom aktuella tidshorisonter bestämma alla ingående tillstånd och ur dessa ta fram en optimal lösning för ett givet produktionsfall vad gäller planering, strategival, produktionsberedning eller styrning av processer [4.16]. Detta gör att det blir av större betydelse att inte nödvändigtvis söka den optimala lösningen i varje situation, utan att ange en möjlig lösning, dvs mera arbeta som en målsökare än som en planerare eller strateg. I ett sådant system är det av största vikt att det finns funktioner som kontinuerligt utvärderar den aktuella situationen om man inte på förhand har full kontroll eller kunskap om vad som sker eller kommer att ske i processen (arbetet). Speciellt gäller det att kunna känna igen händelser som kommer att leda till katastrofala haverier, etc.

Ett område inom AI-tekniken som är under kraftig utveckling är visiontekniken och då speciellt dess användning inom verkstadsindustrin [4.17] [4.18]. Speciella arrangemang har ofta fått tillgripas i beräkningsalgoritmer eller hårdvara (t ex parallella processorer) för att klara de realtidskrav som eventuellt ställs i ett specifikt fall. För att förbättra tekniken söker man nu nya modeller som arbetar i analogi med biologiska visionsystem. Med hjälp av sådana modeller hoppas man kunna få en effektivare struktur på hur en bild skall representeras internt i datorn samt hur en dator skall vara uppbyggd för

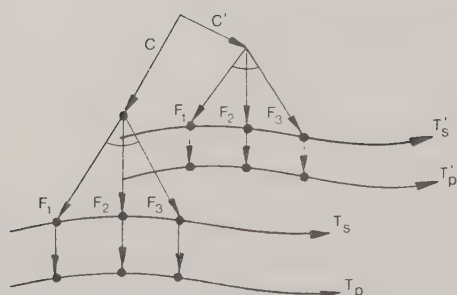


Bild 4.11. Två olika kommandon C och C' genererar olika rörelsesekvenser [4.20].

att få en effektivare bildbehandling.

Att studera den mänskliga hjärnan för att utifrån den skapa modeller över hur olika system skall se ut (planering, lärsystem, motorik, etc) är en ofta använd metod [4.14] [4.19], då de biologiska systemen fortfarande i de flesta fall är helt överlägsna artificiell intelligens.

För att beskriva förlopp inom AI-tekniken brukar man använda sig av OCH/ELLER-diagram. Ett sådant diagram beskriver diskreta tillstånd och man behöver därför göra vissa modifieringar av tillståndbeskrivningen för att kunna använda tekniken på motoriska system, där man skall löpa genom ett kontinuerligt rörelseförlopp. Man kan då låta hela sekvenser av OCH-noder som beskriver rörelser och alternativa ELLER-noder ligga inom stora möjliga arbetsområden [4.20], se bild 4.11.

I bild 4.11 kan man se $T_S = (\dots, F_1, F_2, F_3, \dots)$ som en ingående vektor i rummet. Denna transformeras eller överförs till en utgående vektor $T_P = (\dots, P_1, P_2, P_3, \dots)$ i rummet. I ett styrsystem kan detta ses som att C är ett kommando typ 'LINJÄR RÖRELSE I RUMMET'. T_S är en vektor bestående av interpolationspunkter i ett kartesiskt koordinatsystem, medan T_P beskriver motsvarande vektor i robotens koordinatsystem (t ex ledvinklar). Med denna teknik är det möjligt att bygga upp ett hierarkiskt system av stegvis förfining för en given uppgift på så sätt att den högsta nivåns utsignal blir den näst högsta nivåns

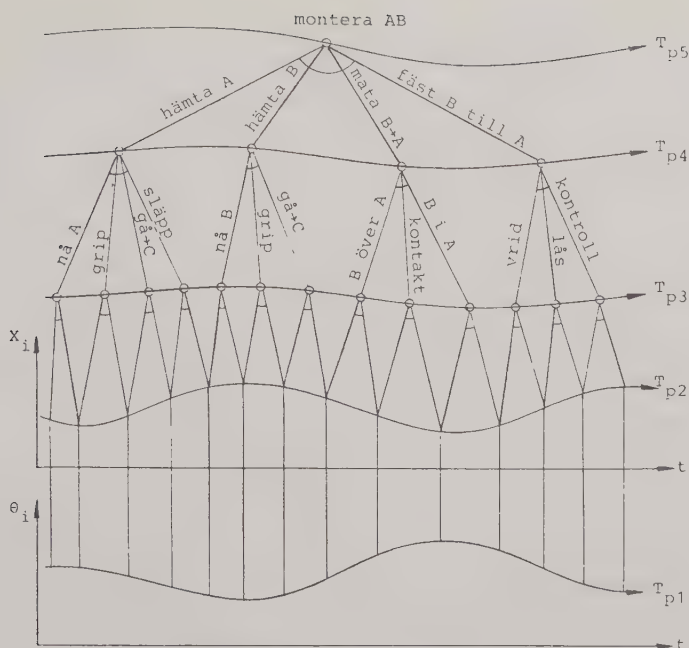


Bild 4.12. En hierarki av stegvis förfining av en monteringsuppgift som producerar ett kontinuerligt OCH/ELLER-träd vilket beskriver ett sensorinteraktivt målinriktat beteende [4.20].

utsignal, dess utsignal nästa nivå's insignal osv ner till den lägsta nivån som innebär den fysiska rörelsen av manipulatern. Bild 4.12 illustrerar detta med en uppgift 'MONTERA AB' [4.20]. Under arbetets gång kommer den högsta nivå's utsignal att generera en serie förenklade uppgifter till nästa nivå som 'HÄMTA A', 'HÄMTA B', 'MATA B TILL A', etc. Den andra nivån tar dessa kommandon som indata och genererar i sin tur elementära rörelsekommandon till nästa nivå i hierarkin som t ex 'GA TILL A', 'GRIP', 'FLYTТА TILL C', 'SLÄPP', etc. Arbetsbeskrivningen kommer på detta sätt att slutligen beskriva fysiska rörelser i rymden som manipulatern utför.

För att hela tiden välja rätt kommando vid varje nivå och tidpunkt under monteringen krävs det att det på samma sätt som vid rörelsegenereringen i det hierarkiska rörelsesystemet finns ett motsvarande hierarkiskt system som återför sensorinformation till varje nivå i rörelsesystemet och som är av vital

betydelse för att den aktuella uppgiften skall kunna utföras. Detta kan på låg nivå innebära hastighetsövervakning, rörelseövervakning (fogföljning), medan man på högre nivåer behandlar mera komplex och generell information för strategi och beslutsval som t ex information från synsystem (vision) eller aktuell status från mataranordningar och periferiutrustning.

Felkorrigeringar som utförs av styrsystemet (t ex vid fogföljning) eller alternativa strategier/handlingssätt (t ex vid val av andra rörelser, annan mataranordning) kan representeras av olika ELLER-noder som i Bild 4.11. Dessa är dock inte utritade i beskrivningen av monteringsuppgiften i bild 4.12 för att inte förlora tydligheten i figuren.

Det eleganta i ett sensoråterkopplat hierarkiskt styrsystem ligger i att de lägre nivåerna kan söka korrigera för smärre förändringar i omgivningen relativt oberoende av övriga nivåer i hierarkin, se bild 4.13. Dessa nivåer kräver också normalt mindre processering än de högre nivåerna, då de algoritmer och samband som ligger nära banstyrningen av manipulatern ofta är väl uttestade och beskrivna matematiskt. Problem av planerande karaktär däremot innebär ofta bestämning och val av en möjlig lösning bland många, där kriterierna för denna lösningssökning inte alltid är helt bestämda. Fördelen med att endast den lägsta nivån ingriper vid förändringar är att systemet kommer att handla och agera snabbt för att söka ta hand om och anpassa sig till en förändring i omvärlden. Om den lägsta nivån misslyckas med att korrigera beteendet av manipulatern med hänsyn till den förändrade situationen, kommer nästa nivå i hierarkin att kopplas in för att söka lösa problemet, osv. Extrema förändringar i omvärlden kan givetvis sätta en gräns för vad ett beskrivet system kan klara av liksom fallet med motsvarande biologiska system. De naturliga begränsningarna sätts av manipulatorerna, sensorerna och de algoritmer som utnyttjas vid återkopplingen av sensordata för styrning av manipulatern.

Genom alltmer avancerad styrning och sensorhantering har den hierarkiska strukturen av system för manipulatorer blivit allmänt använd [4.19] [4.21] [4.22] [4.23] [4.24].

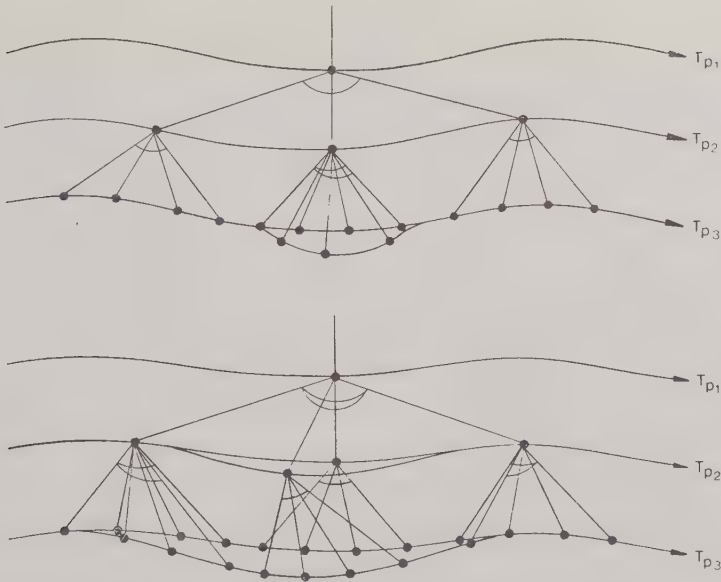


Bild 4.13. En liten förändring i omvärlden kan korrigeras på låg nivå i hierarkin (a) medan större förändringar kan kräva korrigeringar i högre nivåer (b) [4.14].

De fördelar som man främst märker vid ett beskrivet hierarkiskt modulariserat styrsystem är:

Uppdelningen av ett system i lätt identifierbara enheter eller moduler leder i varje steg till en förenkling av ett givet problem. Modulerna skall lösa ett större problem genom att dela upp detta i en mängd delproblem vilka löses av mindre moduler i systemet, strukturerade i en (eventuellt flerdimensionell) hierarki.

Genom modulariseringen och den stegvisa förfiningen av ett givet problem uppnår man följande: Över en viss nivå i hierarkin är man oberoende av vilken manipulator som skall utföra arbetet (så länge manipulatorn har fysiska möjligheter att utföra det aktuella arbetet). Genom detta uppnår man en viss generalitet i systemet, främst beträffande de högre nivåerna som gäller planering, strategival, etc. Även den totala systemstrukturen uppnår generalitet, då man vid styrning av en annan manipulator inte behöver bryta ner eller bygga upp

ett helt nytt styrsystem utan antingen byta ut de nivåer i hierarkin som reellt styr manipulatern i den fysiska miljön och/eller byta ut de transformationsmatriser som överför en rörelse från ett kartesiskt koordinatsystem till robotens koordinatsystem.

I och med att systemet är starkt modulariserat krävs en sträng standardisering beträffande informationsutbytet mellan modulerna och den uppgift som varje modul skall svara för. Detta innebär att systemet som helhet inte kommer att påverkas vid smärre ändringar av en specifik funktion hos ett kommando i en enskild modul så länge modulen håller sig till den bestämda standarden. En standard i datakommunikationen till och från systemet kan också innebära stora fördelar för den yttre informationen. Främst gäller detta styrningen av manipulatern, där en väl definierad standard kan förenkla problemet att kunna styra olika robottyper från samma typ av styrsystem. Den andra sidan gäller programmeringen av robotsystemet, där man genom standard i indata till systemet kan öka förutsättningarna för en utvecklad användning av off line programmeringsteknik och då speciellt utnyttjandet av CAD/CAM-system vid programmeringen av industrirobotar.

En hög standardisering av informationsflödet mellan modulerna i systemet kommer att medföra att införandet av tillägg i funktionen hos enskilda moduler t ex utökning av kommandorepertoaren eller modifiering av algoritmer för sensoråterkopplingen inte kommer att störa den totala bilden av systemet. Dessa modifieringar kan införas i respektive modul om standarden hålls. De förändringar som eventuellt måste göras ligger i de intilliggande modulerna om den aktuella modulens in- respektive utdata kommer att ändras i och med förändringen.

Genom förenklingen av problemet till hanterbara delproblem kommer man att erhålla beräkningsrutiner, styralgoritmer och återkopplingsrutiner som ligger inom hanterbara enheter. Komplexiteten i varje modul kan i stort hållas på en övergripbar nivå oberoende av hela strukturens komplexitet.

4.2 SENSORSYSTEM

För att robotsystemet skall kunna arbeta i en föränderlig omgivning krävs det att det finns information tillgänglig som beskriver de förändringar som uppstår i omgivningen och som är relevanta för den process som roboten medverkar i. Vid bågsvetsning är det speciellt fogföljningsproblematiken och kontroll samt styrning av svetsparametrar som är av stort intresse. Även annan information är av värde t ex kontroll av ingående detaljer och fixtur. Inledningsvis behandlas den principiella uppbyggnaden och användningen av sensorsystem för industrirobotar [4.25] och därefter hur sensortekniken kan utnyttjas för svetsrobotar.

4.2.1 ANVÄNDNING AV SENSORER OCH SENSORSYSTEM I ROBOTSYSTEM

En tillverkningsprocess är till sin natur delvis en stokastisk process, där händelser uppstår slumpmässigt i tiden. En mänsklig operatör upptäcker oftast för tillverkningsprocessen eller sig själv viktiga händelser med hjälp av sina fem sinnen (syn, hörsel, känsel, lukt och smak) och hjärnan. För att utröna om ett tillstånd är önskvärt eller ej krävs en ingående kunskap och kännedom om processen samt en erfarenhetsmässig kunskapsbank om tidigare inträffade händelser och samband som kan associeras till den som utförs. Om detta är möjligt, kan man ställa upp hypoteser eller förväntningar på hur en process skall fortlöpa samt även kunna lägga upp strategier för hur längre förlopp skall utföras. Om tillståndet under processen avviker från det önskvärda, vidtas åtgärder för att rätta till felet genom utnyttjande av processkunskaper, erfarenhet från liknande situationer, manipulatororgan och ingående maskiner.

Antag att man placerar en industrirobot i en tillverkningsprocess, där man normalt har en mänsklig operatör för att kontrollera, styra, utföra och korrigera händelser i en process. Vilka egenskaper bör ett till industriroboten kopplat sensorsystem ha? Man kan skilja ut två artskilda funktioner, nämligen

- bestämning av fel i position och orientering
- bestämning av fel i processen

Nedan skall dessa funktioner beskrivas närmare.

4.2.1.1 BESTÄMNING AV FEL I POSITION OCH ORIENTERING

Med bestämning av fel i position och orientering menas att mäta hur robotens rörelsemönster (egentligen dess arbetspunkt) skall korrigeras för att stämma överens med den verklighet som industriroboten för närvarande arbetar i, dvs att beräkna hur rörelsen skall modifieras med hänsyn till de fluktuationer i toleranser som förekommer i produktionen (dessa kan även uppträda under processen, t ex bågsvetsning, där plåtarna kan krympa eller slå sig under inverkan av värmetillförseln).

Om differensen mellan verklighet och modell är stor, är det lätt att förstå att det kan uppstå problem att fullfölja en arbetsuppgift. Robotsystemet kan då helt enkelt inte hantera den förändrade situationen, främst beroende på att dess arbete är definierat utifrån en rörelse som endast stämmer i en ideal modell. Om däremot roboten kan återkoppla sitt arbetssätt med information från sensorsystem, finns möjlighet till förändring av arbetssättet under processen.

För att minimera differensen mellan verklighet och modell kan arbetsstyckena fixeras i jigger, fixturer eller paletter som är väl definierade i robotens arbetsområde. Metoden är vanligtvis kostsam (åtminstone vid komplexa detaljer), vilket är en av anledningarna till att industrirobotar har liten användning i småserieproduktion.

Ett annat angreppssätt är att låta arbetsstyckena komma in i arbetsområdet på ett mer slumpartat sätt (t ex på ett transportband eller i en låda) som är fallet hos mänskliga operatörer. Om man kan lösa problemet att finna detaljernas position och orientering i robotstationen, skulle denna metod vara betydligt enklare att använda vid småserier, där speciella jigger och fixturer annars skulle göra att kostnaden blir hög räknat per detalj.

De positions- och orienteringskorrigeringar som roboten skall utföra beror i hög grad på typ av arbete som skall utföras. Vid hantering av detaljer eller vid maskinbetjäning kan det röra sig om sökning efter detaljer i ett fördefinierat område. Vid montering av detaljer med snäva toleranser kan det röra sig om finjustering med tryckande och vridande rörelser. Vid processapplikationer som bågsvetsning är det ofta förmågan att följa en fog som är avgörande.

Vid styrning av rörelsen med hjälp av sensorinformation är det av stor betydelse att kunna kontrollera och undvika kollisioner med omgivningen. Kollisioner kan undvikas dels med hjälp av sensorer som reagerar på en för hög kraftpåverkan eller kontaktlösa sensorer typ vision som ser hindren i robotens rörelsebana, dels genom att man definierat förbjudna områden i robotens omvärld som sätter gränser för robotens arbetsområde.

4.2.1.2 BESTÄMNING AV FEL I PROCESSEN

Med bestämning av fel i processen menas t ex inspektion av detaljer, processövervakning samt feltest av de i arbetsprocessen ingående enheterna (maskinerna). Detta skall pågå i reell tid, dvs under arbetsprocessens gång och utan att nämnvärt påverka arbetsförloppet.

Inspektion av detaljer är av väsentlig betydelse genom hela arbetsprocessen. Det kan gälla alltifrån närvarobestämning av detaljer till automatisk inspektion av röntgenfotograferade svetsfogar. Denna typ av inspektion kan ge indikation på om processen fortlöper på ett korrekt sätt med avseende på arbetsstycken, verktyg och maskiner.

Processövervakning innebär en integration av roboten i arbetsprocessen. Om man tar bågsvetsning som exempel, skall en sådan station kunna övervaka och optimera processen som sådan (t ex ström och spänning), men även kunna integrera processen i robotens arbetssätt så att roboten om nödvändigt ändrar svetshastighet eller börjar pendla.

Feltest av ingående komponenter (maskiner) är väsentlig för kvaliteten på det utförda arbetet. Det kan t ex vara att bestämma fel på svetsmaskiner, fixturer, förslitning av komponenter i kritiska delar i en process, etc.

4.2.2 PRINCIPER FÖR BESTÄMNING AV TILLSTAND

Under diskussionen tidigare har olika motiv för användandet av sensorer kommit fram liksom vilka tillstånd som därvid kan vara av intresse för ett robotsystem med sensorinteraktivitet att mäta och ta hand om. Man kan skilja mellan två artskilda mätmetoder, sensorer av kontakt respektive icke kontakt-typ. Kontaktsensorer mäter normalt kraft, vridmoment och fysisk kontakt med föremål (känsl eller tryck). Icke kontaktsensorer mäter närvaro av föremål (avstånd) och visuella bilder. Nedan kommer de olika mätmetoderna att diskuteras närmare.

4.2.2.1 KONTAKTSENSORER

Sensorer under denna rubrik bygger främst på känsl och kraftdetektering. Vilka möjligheter har man att applicera sådana sensorer i ett robotsystem? Man kan skilja ut ett antal principer som

- återkoppling av krafter och vridmoment genom reaktionskrafter i robotlederna
- bestämning av krafter på arbetsbordet
- bestämning av krafter i handleden
- känsl i gripfingrarna

Mätning av krafter och vridmoment i robotlederna kan göras genom att för t ex en robot med DC-motorer mäta strömmen till motorerna som driver robotens armar. Metoden är enkel att använda men ger dålig noggrannhet, främst beroende på okända friktioner och varierande laster och moment vid olika ledpositioner.

Mätning av krafterna på arbetsbordet kan vara en intressant lösning då krafterna på arbetsstycket skall bestämmas som t ex vid montering. Metoden kan innebära vissa fördelar då man vill skräddarsy ett sensorarrangemang för en speciell detalj eller monteringsfixtur och man slipper göra ingrepp på robotarmen.

Mätning av krafterna i handleden är oftast att föredra då sensorn i detta fall ligger nära gripdonet eller verktyget. Sensorarrangemanget placeras normalt vid sista länken mellan roboten och gripdonet/verktyget vilket gör att man med stor säkerhet mäter de tillstånd som är väsentliga för robotens arbete utan alltför stor påverkan från dynamiska krafter som uppstår under robotens arbete.

Känsl i gripfingrarna kan åstadkommas genom att man monterar ett antal kontakt- eller känslensensorer i ett mönster på gripfingrarna. Detta ger möjlighet att kontrollera finmotoriken vid monteringsoperationer samt i en del fall igenkänning av föremål genom det tryckmönster som uppstår vid kontakt.

En metod att anpassa robotens arbetspunkt i rymden till arbetsstycket vid t ex montering är eftergivlighet i handleden. Eftergivlighet i handleden är en passiv egenskap hos fjäderelement som placeras mellan änden på handleden och gripdonet och är alltså inte något egentligt sensorsystem. Tekniken är emellertid intressant och kan ha sitt berättigande i vissa applikationer som t ex montering, där man söker utföra inpassningar av olika slag. En annan funktion som eftergivlighet kan ha är att verka skyddande (att ge efter) då alltför stora krafter uppstår under arbetet, t ex vid oavsiktlig kollision.

4.2.2.2 ICKE KONTAKTSENSORER

Sensorer av icke kontakttyp har sina speciella fördelar i identifiering av detaljer, sökning, visuell inspektion, läges- eller avståndsbestämning av detaljer, etc. De huvudsakliga sensorsystemen under denna kategori är

- optiska sensorsystem
- närhetssensorer

Optiska sensorsystem kan innebära användning av alltifrån enkla fotodioder till avancerade visionsystem. Vision har sin givna plats vid uppgifter som identifiering, inspektion, positions- och orienteringsbestämning (t ex fogbestämning).

Närhetssensorer kan användas för att detektera närvaron av ett föremål utan att vara i kontakt med det. Applikationsområdena kan vara sökning efter ett föremål inom en given volym eller fogföljning vid bågsvetsning. Principer som kan användas är induktans, kapacitans och magnetism. Med t ex magnetism kan även mönster i omgivningen fångas upp vilket i en del fall kan användas på samma sätt som vid vision.

4.2.3 BESKRIVNING AV OLIKA SENSORTYPER

Sensorer eller givare används generellt för att mäta ett tillstånd i en process. Dessa mätdata eller processade data kan sedan bilda underlag för hur processen fortlöper, vilka korrigeringar som skall vidtas, etc. Vilka sensorer och fysikaliska principer man kan använda sig av vid mätning av ett tillstånd beror till stor del på den process man vill observera, vad man vill mäta och hur miljön runt processen ser ut (processens störkänslighet för sensorn och tvärtom). Det sistnämnda är inte minst viktigt då flertalet av de sensorer eller givare som finns tillgängliga idag främst är lämpade för en laboratoriemiljö och ej den miljö som oftast finns vid en arbetsprocess på verkstadsgolvet med rök, smuts, damm, värme, etc.

Traditionellt delar man upp industriroboten med avseende på typ av arbete i hanteringsrobotar, processrobotar och monteringsrobotar. Vilka mätproblem kan uppstå vid dessa arbetsprocesser? Ett gemensamt problem för samtliga servostyrda robotar är att det är nödvändigt att kunna mäta robotens läge (och ofta även hastighet) under det att roboten rör sig. Mätproblemet kan skilja sig från robot till robot beroende på dess kinematik och drivsystem, men normalt gäller det att mäta ett läge linjärt eller angulärt (t ex för-

flyttning av en robotaxel genom utdragning av en cylinder eller rotation i en led).

Ett vanligt problem vid hanteringsrobotar är att man vill söka efter ett föremål inom en på förhand definierad arbetsvolym. Vid hantering av detaljer kan det också vara värdefullt att kunna kontrollera den kraft och det tryck som detaljerna utsätts för både vid gripning och vid kontakt med andra föremål. Om man vill utnyttja roboten vid flexibel automatisering och använda sig av ett minimum av fast periferiutrustning, är det nödvändigt att kunna bestämma detaljers position och orientering i robotstationen.

Vid processrobotar kan en mängd parametrar vara av intresse beroende på den aktuella processen och dess speciella karakteristika. Några parametrar av generell natur är krafter och moment (t ex gradning), acceleration och vibration (t ex slipning), positions- och orienteringsbestämning (t ex bågsvetsning). För bågsvetsning följer även kontroll av parametrar som styrs av strömkällan som ström, spänning och pulsning.

Vid monteringsrobotar finns samma problem som ovan med kraft-, tryck-, moment- och lägesbestämning. Sensorer för montering kräver dock ofta noggrannare mätdata på grund av de vanliga toleransproblem som uppstår vid montering (s k byrålådeproblem vid inpassningar). Sensortekniken vid montering är emellertid intressant även för bågsvetsning i de fall, där roboten även utnyttjas att lägga detaljer i fixturen.

Sammantaget kan sägas att de viktigaste storheterna som behöver mätas vid en robotstation är

- mätning av linjärt läge
- mätning av vinkelläge
- mätning av kraft
- mätning av tryck
- mätning av acceleration och vibration

Samtliga dessa storheter kan mätas med en mängd olika fysikaliska principer. De vanligaste använda principerna är

- induktiva givare
- resistiva givare
- kapacitiva givare
- magnetiska givare
- optiska givare

Vilken princip som skall användas beror i hög grad på den aktuella applikationen, miljön som sensorn skall arbeta i, noggrannhet, etc. Så t ex kräver induktiva närhetsgivare (icke kontakt) att det föremål som man mäter på är av metall, som påverkar induktansen i en oscillator-krets. Magnetiska givare kräver på analogt sätt att det finns ett magnetiskt fält som givaren kan mäta på. Resistiva givare för linjär mätning bygger ofta på någon form av skjutpotentiometer. Dessa är mekaniskt känsliga för snabba ofta återkommande mätningar eller rörelser, varför de bör användas, där mätningar sker mindre frekvent. Dessa exempel visar att det är synnerligen viktigt att studera och ta reda på vilken storhet som skall mätas, miljön vid mätstället och möjliga principer för mätning innan man bestämmer sig för en speciell sensortyp eller sensorsystem.

4.2.4 SENSORER OCH SENSORSYSTEM FÖR BAGSVETSNING

De sensorer och sensorsystem som är utvecklade speciellt för bågsvetsning används för fogföljning och processmätning. Då sensorn mäter nära ljusbågen under svetsning ställs en mängd funktionskrav bland vilka kan nämnas:

- processoberoende med avseende på fogtyp
- objektet skall inte kräva någon extra förberedelse med avseende på sensorn
- informationen skall kunna ge en 3-dimensionell bild av fogens utseende

- mätning och processering av informationen skall utföras i realtid för att möjliggöra t ex fogföljning under svetsning (innebär en samlingsfrekvens på upp till 100Hz enligt diskussionen ovan i avsnittet om styrsystem)
- sensorn skall ha en kompakt uppbyggnad för att inte minska åtkomligheten på objektet
- sensorn skall ha en robust konstruktion som tål den mekaniska och elektriska miljö som råder under svetsning
- sensorn skall vara enkel att använda med avseende på handhavande och programmering

Med tanke på de funktionskrav som ställs på ett sensorsystem för bågsvetsning inser man att användningen av dessa är mycket begränsad med dagens teknik. De metoder och principer för sensorer som kan komma till användning vid bågsvetsning skall studeras nedan. Man kan urskilja tre artskilda metoder, nämligen

- taktila sensorer
- ljusbågssensorer
- optiska sensorer

Taktila sensorer är enkla och robusta, men har den nackdelen att de inte ger någon information om hur fogens geometri ser ut. Metoden bygger på att man låter en mekanisk mätprob släpa längs fogen eller för in en prob i fogen vid på förhand bestämda tidpunkter. I och med detta kan man få problem vid vissa geometrier och skarpa hörn, t ex parallaxfel.

Ljusbågssensorer bygger på principen att spänningen över bågen är proportionell mot båglängden. Genom att mäta spänning och ström kan man genom detta förhållande styra svetspistolen över en konstant höjd över arbetsstycket. Om svetspistolen eller bågen förs från sida till sida, kan man erhålla en höjdprofil av den yta som man går över. Om man på analogt sätt gör en oscillerande rörelse längs en svetsfog, kan man erhålla en profil av fogens utseende. Den oscillerande rörelsen kan åstadkommas antingen mekaniskt eller magnetiskt. Fördelen med metoden är att man mäter direkt vid svetspunkten. Det krävs få extra komponenter på svetspistolen och man får en kontinuerlig kontroll av rörelsen under svetsning. Nackdelen är att det som regel krävs en

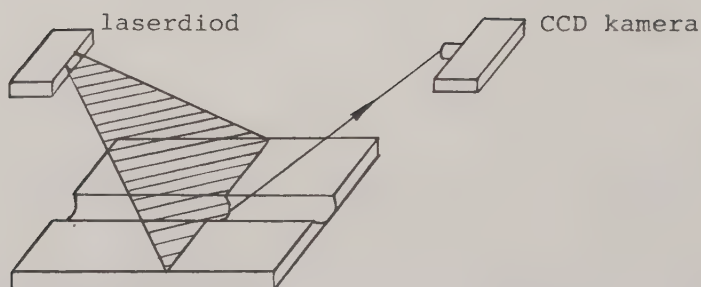


Bild 4.14 Principiell funktion av triangulering med strukturerat ljus.

väl definierad startpunkt för att komma in rätt på svetsfogen. Noggrannheten är beroende på fogtyp och den geometriinformationen som erhålles. Genom en kombination av flera sensortyper kan en förfinad information erhållas.

De optiska system som ger 3-dimensionell information angående fogen använder principen att mäta tiden för ljusets väg mellan sändare och mottagare eller triangulering med strukturerat ljus. Mätning av tiden för ljusets väg mellan sändare och mottagare använder sig av pulserande eller modulerat laserljus. Praktiska försök har visat stora realtidsproblem vid robottillämpningar [4.26]. Triangulering med strukturerat ljus har däremot visat sig mycket användbart. Man låter här en ljusstråle projiceras snett mot svetsfogen och erhåller en profil av fogens utseende. En CCD-kamera (Charge Couple Device) används sedan för att få en bild av fogens utseende, se bild 4.14. Med hjälp av filtreringsteknik kan man erhålla en 3-dimensionell bild av fogen för tex fogposition, fogvolym, fogbredd, etc. En variant av denna teknik är avståndsmätande system, där man låter en laser projicera en ljusstråle mot fogen. En positionskänslig fotodetektor mäter sedan reflektionen från arbetsstycket och ger ut en ström som är proportionell mot avståndet till mätpunkten, se bild 4.15. För att få information om fogens profil måste laserstrålen svepas över fogen. Avståndsmätande system är ett av de fåtal system som nått praktisk användning vid robotsvetsning [4.27] och används

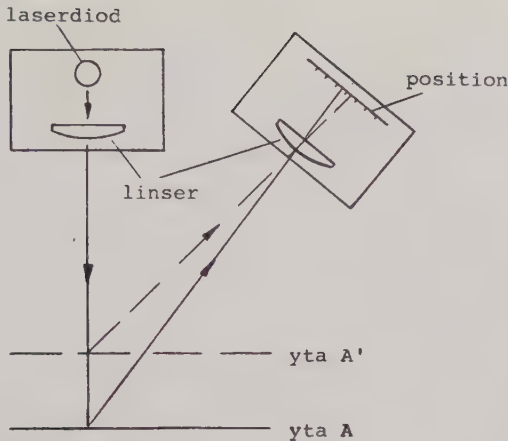


Bild 4.15 Principiell funktion av avståndsmätande optiskt system.

för att söka fogens start- och slutposition.

Sensorsystem baserade på principen triangulering med strukturerat ljus är en vanligt förekommande metod vid forskning inom området. Sensorn placeras så att man mäter på fogen ca 20mm före bågen. Typiska data som kan erhållas med metoden är en upplösning av storleksordningen 0.1-0.2mm beroende på optik och med en samplingsfrekvens om 5-20Hz. Den 3-dimensionella information som kan erhållas är fogöppning, spalt, fogvolym, fogvinklar, häftsvetsar, etc. För att kunna använda kamera vid den miljö som råder använder man sig av filtertechnik så att man endast tittar på ett fönster kring det våglängdsområde som laserljuset har. Typiskt är 830nm med ett fönster på 825-835nm. I och med den rika informationen anses metoden vara anpassningsbar till ett flertal fogtyper. En nackdel är dock att det i robotsammanhang krävs en extra rörelseaxel för att sensorn kan vrida sig runt svetspistolen för att följa med fogens orienteringsriktning [4.28] [4.29] [4.30]. Genom den stora informationen söker man utifrån fogprofilen, smältans utseende och svetsfogens utseende efter bågen skapa processmodeller som gör att man utifrån kunskapsbaserade system skall kunna erhålla eller styra svetsparametrar under svetsning för att uppnå ett korrekt svetsresultat [4.31] [4.32] [4.33].

Den ovan refererade tekniken är möjlig att använda under svetsoperationen. En enklare metod är att använda sig av ett visionsystem som ger en 2-dimensionell information om svetsfogens position. Ett sådant system måste arbeta i två omgångar, där den första innebär att mäta upp var fogen ligger och den andra innebär att utföra själva svetsningen med hjälp av de positionsdata som man erhåller under den första körningen med sensorn inkopplad. Med den beskrivna tekniken finns det idag system i praktisk drift med en mätupplösning på 0.4mm och en samplingsfrekvens på 17Hz [4.34].

Som nämnts tidigare är det möjligt att utnyttja ljusbågen som sensor genom att mäta ström- och spänningsvariationerna över fogen. Metoden är praktiskt användbar idag och ger för TIG- och MIG-svetsning en upplösning av 0.25mm i avståndsmätningen. Ett problem med metoden är oscilleringen av bågen vid robotsvetsning. Ett sätt att lösa detta har varit att montera en speciell enhet på robotens handled som utför en snabb oscillerande rörelse [4.35].

Genom infraröd sensorteknik kan man mäta smältans temperaturgradienter. Den kan användas för att bestämma svetsprocessens förlopp och karaktär. Försök har visat att det bildas en assymetrisk temperaturfördelning vid geometriska variationer (avvikelse från fogen). Vidare har man kunnat påvisa en bestämning av inträngningen utifrån isothermkurvor från givna materialdata, fogtyp och isothermkurvornas utseende [4.36].

En annan intressant användning av visionteknik är att utnyttja den för automatisk kvalitetskontroll av svetsar [4.37]. I detta fall utnyttjas visiontekniken för att avsyna röntgenfotograferade svetsar med avseende på blåsor eller andra defekter.

4.2.5 FUNKTIONELL UPPDELNING AV SENSORINFORMATION

Enligt diskussionen i kapitel 1 önskar man uppnå en programmeringsmetodik som vid småserietillverkning är av arbetsbeskrivande karaktär. Denna metodik kräver, då robotens omgivning inte till alla delar är känd på förhand, fortlöpande information om den aktuella situationen på arbetsplatsen. Vilken typ av information som robotsystemet behöver för att klara en given situation

beror helt och hållet på hur den aktuella arbetsstationen ser ut, typ av arbetsuppgift, etc. För ett generellt robotsystem gäller det därför att kunna anpassa och utnyttja ett stort antal sensorsignaler samt att det skall vara möjligt att bygga ut systemet med nya sensorer efter behov. Då antalet sensorer och deras funktion i ett robotsystem kan vara mycket stort är det lämpligt att gruppera sensorinformationen på så sätt att det är relativt enkelt att utnyttja sensorinformationen vid bestämning av styraktiviteter. För detta ändamål utnyttjas en funktionell uppdelning av sensorinformationen. Denna kan se ut som

- identifiera detalj
- bestämma läge för detalj
- inspektera detalj
- kontrollera manipuleringen
- kontrollera processen

där detalj syftar på en ingående komponent eller ett avsnitt av ett objekt (t ex svetsfog). Nedan skall dessa funktioner beskrivas närmare var för sig.

4.2.5.1 IDENTIFIERA DETALJ

Som nämnts tidigare kan man via mataranordningar eller automater se till att rätt detalj alltid kommer in i robotens arbetsområde på ett riktigt sätt. Detta kräver emellertid en noggrann planering av robotstationen för varje tillverkningsprocess innan en produktion kan starta. En strävan är att kunna utnyttja industriroboten vid småserier. Detta innebär att kostnader för planering och kringutrustning måste reduceras för att uppnå lönsamhet även vid kortare serier.

Om man enbart inriktar sig på vilken detalj, som skall delta i arbetsprocessen, är det nödvändigt för robotsystemet att skaffa sig kunskap om dels vilken detalj den skall arbeta med, dels val av detalj bland flera inom robotens arbetsområde. Vid bågsvetsning kan identifieringsfunktionen användas för att bestämma aktuell fogtyp. Om informationen kopplas till ett kunskapsbaserat system, kan man sedan genom detta erhålla svetsparametrar som är anpassade

för den aktuella fogen.

Detta innebär att om man kan förse robotsystemet med ett flexibelt identifieringssystem, är det möjligt att spara stora kostnader och tid vid planering och riggning av en ny tillverkningsserie, där det ingår industrirobotar. Kostnaden för ett identifieringssystem kan därför väl motiveras, om seriestorlekarna är små och om man önskar hög beredskap för att snabbt kunna sätta in en ny produktion i robotstationen.

4.2.5.2 BESTÄMMA LÄGE FÖR DETALJ

Att bestämma ett läge för en detalj innebär att detaljens position (t ex tyngdpunktens rymdkoordinater av en detalj eller svetsfogens startpunkt) och orientering (t ex huvudtröghetsaxlarnas riktningar för en detalj eller svetsfogens riktning) bestäms. Traditionellt utförs lägesbestämningen rent mekaniskt genom speciella matar- och orienteringsanordningar eller fixturer. Sådana anordningar ser då till att en speciell detalj eller fog kommer i definierad position och orientering. Dessa anordningar har ofta den nackdelen att de är svåra att ställa om för andra detaljer (åtminstone om de geometriska variationerna är betydande) samt att processen kan medföra att förändringar uppstår under själva processen (t ex deformationer i plåtar under svetsning). Detta gör att riggningstiden blir lång och dyrbar om antalet varianter är stort. Om man däremot har mindre seriestorlekar, önskar man korta riggningstider för att inte kostnaden för varje serie och därmed även varje detalj skall bli för hög. Lägesbestämningen bör för mindre serier därmed vara både universell och flexibel för att passa flertalet detaljer.

4.2.5.3 INSPEKTERA DETALJ

Med inspektion av en detalj menas den kontroll som normalt sker vid manuell tillverkning dels före en process, dels efter det att en process är utförd. Kontrollen syftar dels till att inga detaljer skall skada processen, dels till att detaljerna är korrekta från processen på så sätt att de föreskrivna toleranserna hålls. Kontrollen före processen kan bestå av processrelaterade

egenskaper som t ex fogberedning och bortsortering av defekta komponenter. Efterkontrollen består av en kvalitetskontroll så att detaljerna har föreskrivna toleranser och att processen utförs med önskat resultat. Genom detta är det möjligt att förebygga haverier, upptäcka slitage på komponenter eller fixturer, korrigera processparametrar, etc. Allt detta sammantaget gör detaljinspektionen synnerlig viktig vid automatisering av en arbetsstation, där detta arbete tidigare till en del ofta utförts omedvetet av mänskliga operatörer.

4.2.5.4 KONTROLLERA MANIPULERINGEN

Manipulering av arbetsstycken eller verktyg kan innebära en mängd olika deloperationer som positionering, orientering, fogföljning, gripning, passning, iskruvning, etc. I en omgivning som är noggrannt definierad är det möjligt att använda sig av en industrirobot utan sensorer för att utföra dessa operationer genom väl definierade rörelsemönster. Vid användning av industriroboten, där arbetsförhållandena inte kan kontrolleras till alla delar krävs att robotsystemet har tillgång till sensorinformation som bestämmer robotens omvärld. Genom detta kan robotsystemet söka förutse händelser i omgivningen och utifrån detta göra korrigeringar i arbetsmönstret.

Genom detta förändras programmeringsmetodiken från att vara rörelseinriktad till att bli uppgifts- eller målinriktad (jämför med arbetsbeskrivningen för ett arbete som en mänsklig operatör utför). Detta innebär att roboten vad gäller rörelsemönstret i princip kommer att bli självprogrammerande, dvs man kan inte på förhand veta vilken banrörelse som roboten kommer att ta vid ett givet tillfälle. För att kunna kontrollera manipuleringen på ett riktigt sätt behöver roboten tillräcklig information om robotens omgivning och om den aktuella processen samt kunskap att ta emot och behandla aktuell sensordata under manipuleringen för att en strategi skall kunna läggas upp som leder till att arbetsuppgiften utförs på ett riktigt sätt, vad gäller slutresultatet. I och med att man på förhand inte känner den exakta rörelsen är det av största vikt att man kan kontrollera rörelsen med avseende på kollision med omvärlden.

4.2.5.5 KONTROLLERA PROCESSEN

Vid all automation av processuppgifter är det viktigt att man har möjlighet att styra och kontrollera processen med avseende på det resultat man vill åstadkomma. För bågsvetsning gäller detta i högsta grad, där man utifrån aktuell fog och de störningar som kan uppstå vill kunna styra processen så att man erhåller ett korrekt svetsresultat för aktuell fog. Vid val av svetsparametrar bör kunskapsbaserade system (expertsystem) komma till användning för att minimera inkörningstiden vid nya objekt.

4.3 SVETSPROCESSEN VID AUTOMATISERAD SVETSNING

Vid automatiserad svetsning ställs stora krav på svetsutrustningen för att processen skall kunna fortgå utan avbrott och med önskat resultat, speciellt som processen skall kunna fortgå utan mänsklig tillsyn. Några av de krav som kan och måste ställas på en robotstation för robotsvetsning vid småserier är:

- säkerhet mot allvarliga haverier som kan skada människor och maskiner
- tillförlitlighet hos svetsutrustningen som måste vara extremt hög vid användning tillsammans med robotar på grund av dessas höga bågtidfaktor jämfört med manuell svetsning
- noggrannhet och reproducerbarheten i svetsdata (t ex ström och spänning) för att säkerställa det svetsresultat man förväntar sig
- stabil och sprutfri ljusbåge över stora strömområden för svetsning under olika förhållanden (t ex tunnplåt som kan ställa krav på låga strömmar)
- användbarhet för ett flertal metalliska material gäller speciellt för flexibel produktion av småserier
- säker svetsdatabestämmning vid varierade svetsförhållanden (material, fogtyper) och möjlighet till styrning av parametrar från stationens styrsystem
- säkerhet att tända ljusbågen vid svetsstart

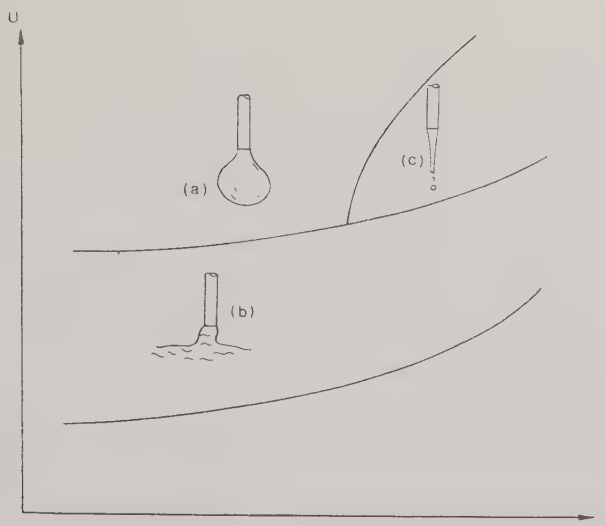


Bild 4.16 Globulär överföring (a), kortbåge (b) och spraybåge (c) vid MIG/MAG-metoden.

Vid traditionell robotsvetsning har man normalt använt sig av MIG/MAG-metoden som har en rad egenskaper som gör den lämpad för automatisk svetsning som en kontinuerlig tillförsel av tillsatsmaterial, eliminering av slagg genom skyddsgas, svetsning i många positioner och hög produktivitet. För att få en stabil båge reglerar man trådmatningen så att man erhåller en konstant båglängd genom att spänningen över bågen är proportionell mot båglängden. Denna reglering kan ske över ett mycket stort strömområde. Emellertid gäller att man för MIG/MAG-metoden och strömmar över en kritisk nivå erhåller en sprutfri materialöverföring som ger en spraybåge med små fina droppar mot smältan. Strömmar under en kritisk gräns ger kortbåge eller globulär överföring med sprut och oregelbunden överföring, se bild 4.16. Vid svetsning av tunnplåt, lägessvetsning och värmekänsliga material kan man emellertid vara tvungen att arbeta under denna kritiska gräns.

En möjlighet att komma runt problemet är att använda sig av en pulsad MIG/MAG-svetsning med lågström för globulär överföring och högström för sprayöverföring som ger en syntetisk spraybåge. Svårigheten är att sätta upp parametrar för en given trådmatning så att pulsamplituden och pulslängden

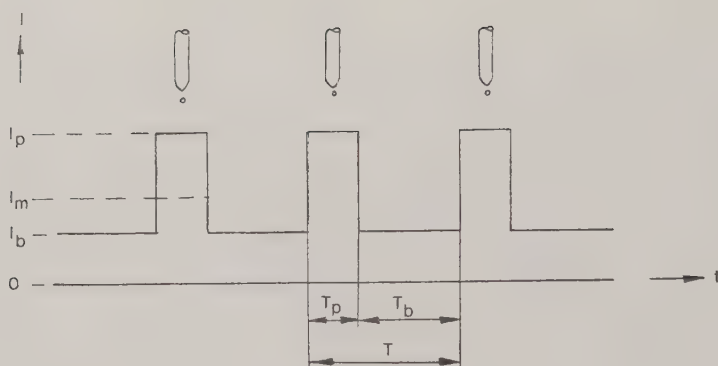


Bild 4.17 Pulsutseende vid SPM.

tillsammans överför minst en droppe under varje puls, se bild 4.17 och att medelströmmen måste regleras så att avbränningen motsvarar trådmattningen för att ge en konstant båg längd. I praktiken är detta ett svårt problem på grund av de variationer som uppstår i trådmattningsmekaniken.

På senare tid har man empiriskt kunnat finna samband som gör att man får en koppling av pulsparametrarna och trådmattningen [4.38]. Metoden kallas Synergic Pulsed Mig welding (SPM) och har främst utvecklats industriellt genom tillkomsten av högfrekventa transistoriserade strömkällor som kan ge en stabil och kontrollerad pulsad svetsström [4.39].

Den generaliserade SPM ger att både bakgrundsströmmen och antingen pulslängden eller pulsfrekvensen skall varieras proportionellt mot trådmattningen under förutsättning att den överlagrade pulsströmstyrkan hålls konstant. I och med detta har man en direkt möjlighet att automatiskt styra svetsparametrarna efter de variationer som uppstår i trådmattning och som annars ger upphov till en instabil överföring. Genom den kontroll man har över svetsprocessen vid SPM uppnår man fördelar för att klara de krav som ställs på svetsutrustningen enligt ovan. Bland dessa kan nämnas enkel inställning genom det kopplade sambandet mellan parametrarna, god överföring av material vid låg medelström, stabilitet vid ändringar i trådmattning, svetsning av många metaller och god noggrannhet vid lägessvetsning.

Mycket arbete återstår för att ta fram optimala svetsparametrar för olika situationer (t ex material, tråddiametrar, fogtyp), där man uppnår exakt en droppe per puls, eftersom man vid efterföljande droppar i samma puls får en mindre effektiv spraybåge. Andra aspekter att ta hänsyn till vid krav på noggranna toleranser är grundmaterialets påverkan på ljusbågen på grund av variationer i joniseringsenergi, som kan göra att bågen viker av från det önskade läget.

4.4 SYSTEMINTEGRATION

För att uppnå en hög flexibilitetsgrad krävs en noggrann planering av de ingående delarna och hur dessa skall samverka i en arbetsstation för robotiserad svetsning. Begreppet flexibilitet innebär här en mängd faktorer som tillsammans bestämmer svetstationens användbarhet och lönsamhet för den avsedda produktionen. Bland dessa kan nämnas

- svetsprocessen som skall användas (MIG/MAG/TIG) och hur denna kan anpassas till den aktuella produktionen
- arbetsområdet som begränsar objektens storlek. Arbetsområdet kan utvidgas genom utnyttjande av lägesställare, montering av roboten på åkdon eller montering av roboten upp och ned
- åtkomligheten som bestämmer objektens komplexitet och fogarnas placering. Åtkomligheten kan ökas genom att hålla volymen kring svetspistolen och handleden till ett minimum eller genom en god orienteringsmöjlighet hos robot eller objekt
- kapacitetsutnyttjandet av robotsvetsstationen som tillsammans med automationsgraden (möjlighet till produktion med begränsad bemaning) bestämmer hur produktionen kan ändras efter marknadsbehovet
- handhavandet vid programmering och inkörning av nya och återkommande produkter bestämmer lönsamheten för småserier i stationen
- objektets och fixturens sammanlagda vikt

Utifrån ovanstående faktorer som styr och begränsar flexibiliteten ser man att ju större svetsproduktion man har för en likartad produktflora, desto större möjligheter har man att begränsa och specialisera stationens produktions-

område till förmån för högre seriestorlekar vilket ger mindre kostnader i samband med handhavande. Ett flexibelt utnyttjande innebär emellertid att man skall ha möjlighet till blandad produktion i stationen. En faktor som kommer att få stor påverkan vid småserieproduktion är kapacitetsutnyttjandet och lönsamheten i och med en ofta återkommande och tidsödande programmering och inkörning vid robotstationen för varje ny produkt eller tillverkningsserie. Önskemål att reducera seriestorleken vid tillverkningen för återkommande produkter beror på de kostnader som uppstår i lagerhållning av förädlade produkter samtidigt som man genom denna reduktion får ett ökat genomflöde av produkter i tillverkningskedjan. Ett avgörande hinder är framförallt den tid som åtgår för omställning av tillverkningen, även då det gäller återkommande produktion.

Man har kunnat påvisa genom simuleringar av layout och produktflöde och senare genom praktiska försök att det är möjligt att utnyttja en robotsvetsstation för en produktflora om upp till 50 olika produkter [4.40] [4.41]. För en station bestående av en robot och tre arbetsplatser (två för svetsning och en för förberedelser) har man vid en svets tid om 5-30 min, ladd- och urladdningstid om 3 min vardera, fixturbytestid om 20-60 min kunnat komma ner i återkommande seriestorlekar på i storleksordningen 10-15 detaljer vid återkommande tillverkning.

Problem som är relaterade till småserieproduktion berör speciellt krav på noggranna mått på ingående komponenter och fixturer för att underlätta igångkörningen av olika serier av samma detalj. Detta har speciellt uppmärksamats i och med att man ofta varit tvungen att korrigera svetsrörelserna vid uppsättning av en serie. Av samma skäl krävs en hög utbildningsnivå på handhavande och programmering eftersom dessa är en ofta återkommande och tidsödande procedur liksom att man tar speciell hänsyn till säkerhetsfrågor då en stor del av olyckor i samband med industrirobotar inträffar vid service, programmering eller annat handhavande då normala skyddssystem är frånkopplade [4.42] [4.43]. För fogar med åtkomlighets- eller orienteringsproblem får man söka nya konstruktionslösningar på detaljen eller utföra manuell svetsning.

Exempel på hur programmeringstiden om en total tid av 40 timmar kan fördela sig över olika aktiviteter är 8 timmar för svetsföljd och svetsparameterar, 16 timmar för robotprogrammering och 16 timmar för test och korrigering av programmet. Detta innebär 4 min/programmerad punkt eller 41 min/svets. Vid uppsättning av en återkommande produktion åtgick normalt 1-4 timmar i ställtid beroende på vilka korrigeringar som krävs i programmerade punkter vid svetsfogarna [4.44]. Enligt [4.45] åtgår 20-30 min programmeringstid för 1 min svetsning med en van programmerare.

Av ovanstående ser man att det idag finns tekniker tillgängliga som gör det möjligt att utnyttja robotsvetsning på ett flexibelt sätt i små återkommande serier. För att man skall kunna komma ner till småserier för den totala produktionen av en produkt krävs att man tar stor hänsyn till hur utrustningen skall utnyttjas. Man kan här konstatera att det finns en stor potential i reducering av programmeringstiden, inplaceringen av ett objekt i svetsstationen och val av svetsföljder och svetsparameterval genom utnyttjande av off line programmeringstekniken och då med hjälp av CAD/CAM-teknik.

Genom off line programmering av roboten kommer man automatiskt in på fogföljnings- och fogsökningsproblematiken som, speciellt för tunnplåt, innebär stora praktiska svårigheter idag både vad gäller sensorer och återkoppling av rörelseinformation till styrsystemet. På processen ställs motsvarande krav genom styrbara strömkällor som har goda svetsegenskaper och då främst pulsade transistoriserade strömkällor med synergieffekt.

4.5 REFERENSER

- 4.1 J. Denavit, R. S. Hartenberg, A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices, Journal of Applied Mechanics, June 1955, pp 215-221.
- 4.2 C. S. G. Lee, Robot Arm Kinematics, Dynamics and Control, IEEE Computer, Dec 1982.
- 4.3 C. S. G. Lee and M. Ziegler, A Geometric Approach in Solving the Inverse Kinematics of PUMA Robots, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago USA.

- 4.4 D. R. Myers and D. F. Gordon, Kinematics Equations for Industrial Manipulators, The Industrial Robot, sept 1982, pp 162-165.
- 4.5 L. van Aken and H. van Brussel, Software for Solving the Kinematic Problem for Robot Manipulators in Real Time, Advanced Software in Robotics, Holland, 1983.
- 4.6 L. Mancini, N. Nicolosi, S. Scarselli and G. Zoly, A New Approach to Coordinate Transformation: Parallell Computation, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago USA.
- 4.7 G. Duelen, J. Held and U. Kirchhoff, The Inverse Kinematic Problem in Realtime Application, Computer Based Factory Automation (Proc 11th Conf on Production Research and Technology), May 1984, Pittsburgh Pennsylvania USA.
- 4.8 ASEA Programming Manual IRb6/2, June 1984, ASEA Robotics inc.
- 4.9 H. I. El-Zorkany, Automatic Location Correction in Off-Line Programming of Industrial Robots, Proc of the 14th Int Symposium on Industrial Robots, 1984, Gothenburg Sweden.
- 4.10 W. A. Gruver, et al, Evaluation of Commercially Available Robot Programming Languages, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago USA.
- 4.11 C. Blume and W. Jakob, Design of the Structured Robot Language (SRL), Advanced Software in Robotics, Holland, 1983.
- 4.12 R. P. Paul and V Hayward, Robot Control and Computer Languages, Theory and Practice of Robot and Manipulators (Proc of RoManSy '84), Kogan Page London 1985.
- 4.13 R. A. Volz, T. N. Mudge and D. A. Gal, Using ADA as a Robot System Programming Language, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago USA.
- 4.14 J. S. Albus, Mechanisms of Planning and Problem Solving in the Brain, Mathematical Biosciences 45:247-293 (1979).
- 4.15 W. B. Gevarter, An Overview of Artificial Intelligence and Robotics, Vol I and II, NASA Technical Memorandum 85836, June 1983, Washington D.C., USA.
- 4.16 J. Hatvany and F. J. Lettner, The Efficient Use of Deficient Knowledge, Annals of CIRP Vol 32/1/1983.

- 4.17 B. K. P. Horn, Artificial Intelligence and the Science of Image Understanding, Computer Vision and Sensor-Based Robots (Edited by George G. Dodd and Lothar Rossol), Plenum Publ. Co., 1979.
- 4.18 D. Mott, Development Tool for Artificial Intelligence in Vision, Sensor Review, Jan 1985.
- 4.19 G. N. Saridis, Toward the Realization of Intelligent Controls, Proceedings of the IEEE, Vol 67, No 8, Aug 1979.
- 4.20 J. Albus et al, Control Concepts for Industrial Robots in an Automatic Factory, SME Technical Paper MS77-745, Dearborn, Michigan 1977.
- 4.21 S. I. Novatchenko, V. A. Pavlov and E. I. Yurevich, Special Modular Software System of Sensitized Robot Control Computer, Mechanism and Machine Theory, Vol 16 pp 41-48, Pergamon Press Ltd 1981.
- 4.22 T. L. Turner, J. J. Craig and W. A. Gruver, A Microprocessor Architecture for Advanced Robot Control, Proc of the 14th Int Symposium on Industrial Robots, 1984, Gothenburg Sweden.
- 4.23 J. S. Albus et al, Hierarchical Control for Robots in an Automated Factory, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago USA.
- 4.24 M. Shneier et al, Robot Sensing for a Hierarchical Control System, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago USA.
- 4.25 G. Bolmsjö, Sensorinteraktiva Styrssystem för Industrirobotar, Mekanisk Teknologi, LTH, Lund 1984 (del B i STU-rapport 82-4117, Flexibelt Robotssystem för Montering och Verktygsbärande Uppgifter).
- 4.26 Z. Smati, D. Yapp and C. J. Smith, Laser Guidance Systems for Robots, Proc 4th Int Conf on Robot Vision and Sensory Controls, 1984, London UK.
- 4.27 Y. M. Yew, Laser and Vision Systems in Robotic Applications, Proc Int Conf on Advances in Manufacturing, 1984, Singapore, IFS(publ) UK.
- 4.28 P. Drews and G. Starke, Development Approaches for Advanced Adaptive Control in Automated Welding, Automation and Robotisation in Welding and Allied Processes, Proc of the Int Conf, Strasbourg, France, 1985.
- 4.29 G. Begin et al, Third Generation Adaptive Robotic Arc Welding Unit, Automation and Robotisation in Welding and Allied Processes, Proc of the Int Conf, Strasbourg, France, 1985.

- 4.30 T. Bamba et al, A Visual Seam Tracking System for Arc-Welding Robots, Proc of the 14th Int Symposium on Industrial Robots, 1984, Gothenburg, Sweden.
- 4.31 J. E. Agapakis, K. Masubuchi and N. Wittels, General Visual Sensing Techniques for Automated Welding Fabrication, Proc of the Int Conf on Robot Vision and Sensory Controls, 1984, London, UK, IFS(publ) Ltd.
- 4.32 N. R. Corby Jr, Machine Vision Algorithms for Vision Guided Robotic Welding, Proc of the Int Conf on Robot Vision and Sensory Controls, 1984, London, UK, IFS(publ) Ltd.
- 4.33 J. E. Agapakis, N. Wittels and K. Masubuchi, Automated Visual Weld Inspection for Robotic Welding Fabrication, Automation and Robotisation in Welding and Allied Processes, Proc of the Conf, Strasbourg, France, 1985.
- 4.34 I. Masaki, M. J. Dunne and H. Toda, Vision Guided Robot System for Arc Welding, Robot Vision, International Trends in Manufacturing Series, IFS(publ) UK 1983.
- 4.35 G. E. Cook, Position Sensing with an Electric Arc, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago, USA.
- 4.36 N. H. Madsen and B. A. Chin, Automatic Welding: Infrared Sensors for Process Control, Computer Based Factory Automation (Proc 11th Conf on Production Research and Technology), 1984, Pittsburgh, Pennsylvania, USA.
- 4.37 K. Inoue and M. Sakai, Recognition Method for Monitoring Weld Result - Fundamental Study on Automation of Inspection, Automation and Robotisation in Welding and Allied Processes, Proc of the Int Conf, Strasbourg, France, 1985.
- 4.38 M. Amin, Synergic Pulse MIG Welding, Metal Construction, June 1981.
- 4.39 M. Jilong and A. R. Louis, The Industrial Implications of Improved Metal Transfer, The Int Conf on Quality and Reliability in Welding, 1984, Hangshou, China.
- 4.40 P. S. Monckton, R. W. Hawthorn and P. M. Hutton, A Robot Welding Cell for Small Batch Production, Proc Int Conf on Advances in Manufacturing, 1984, Singapore, IFS(publ), UK.
- 4.41 P. S. Monckton, R. W. Hawthorn and R. Jones, Simulation of a Robotic Welding Cell to be used in Small Batch Production, Proc of the British Robot Association Annual Conf, 7th, 1984, UK.

- 4.42 R. G. Thorne and J. E. Middle, The Factors Effecting the Successful Implementation of a Robotic Arc Welding System in Batch Manufacturing, Developments in Robotics 1983, Compiled by Brian Rooks, IFS(publ), UK, 1983.

- 4.43 P. K. Wadsworth, R. G. Thorne and J. E. Middle, Feasibility Studies for Small Batch Robotic Arc Welding, Proc of the 13th Int Symposium on Industrial Robots, 1983, Chicago, USA.

- 4.44 R. W. Richardson et al, The Feasibility of Robotic Arc Welding of a Low Volume Sheet Metal Fabrication, Proc of th 13th Int Symposium on Industrial Robots, 1983, Chicago, USA.

- 4.45 Personliga samtal med P-A Larsson, B. Pekkari, ESAB AB, Laxå.

5 UPPBYGGNAD AV ETT STYRSYSTEM FÖR SVETSROBOTAR

För att möta de krav som ställs i kravspecifikationen och samtidigt beakta ovanstående diskussion och argumentering kring hierarkiska och modulariserade robotsystem presenteras en strukturell uppbyggnad av ett sensorinteraktivt styrsystem för flexibel automatisering.

Vid en modulariserad uppbyggnad kommer följande tre enheter att bilda huvudmodulerna i systemet [5.1], se bild 5.1:

- Programmodul
- Produktmodul
- Realtids styrsystemmodul

Programmodulen skall ge stöd i programmeringen i olika former. De egenskaper som är viktiga är snabb och effektiv editering av ett användarprogram, simulering av användarprogram samt off line och on line programmeringsmöjlighet. I programmodulen finns även de enheter i robotsystemet som bestämmer hur språket, dess syntax och grammatik skall se ut. Detta gör att stor hänsyn måste tas till hur programmodulen byggs upp med hänsyn till framtida möjligheter att lägga in nya kommandon eller andra tillägg i användarspråket. De kommandon som ges i ett användarprogram skall inte nödvändigtsvis behöva anges explicit (t ex explicita rörelsekommandon, där xyz-koordinaterna anges numeriskt eller implicita rörelsekommandon, där symboliska namn eller relationer anger hur rörelsen skall läggas ut), utan symboliska namn som har sin definiering någonstans i databasen skall kunna användas, t ex

WELD JOINT_1 FROM A TO B, OPTIMIZE WELDPARAMETERS

Kommandot ovan kan generera en korrekt svetsrörelse i rummet genom information som finns tillgänglig om hur JOINT_1 ser ut, var punkterna A och B finns, JOINT_1:s läge och orientering i robotens omgivning, strategi för hur

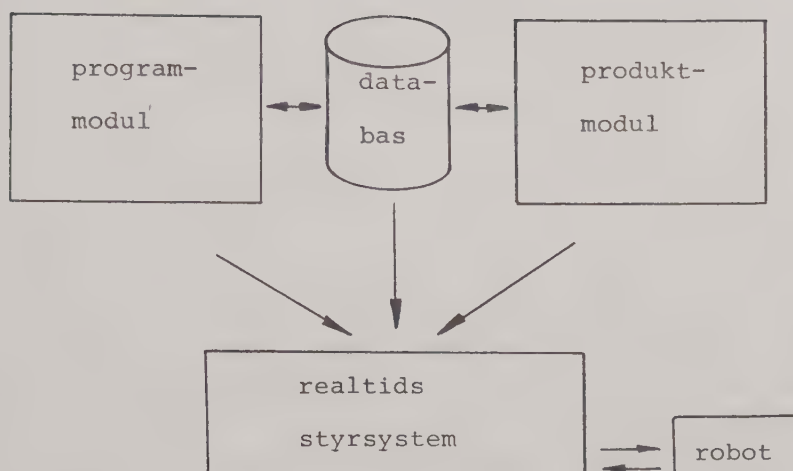


Bild 5.1 Huvudmoduler i ett flexibelt sensorinteraktivt styrsystem [5.1].

rörelsen skall läggas ut med korrekt pistolvinkel och eventuell koordinerad rörelse av lägesställare för att säkerställa processen samt slutligen styra ut korrekta svetsparametrar för den aktuella fogen med ledning av den information som finns tillgänglig (material, skyddsgas, fogtyp, svetsläge, svetstråd, etc).

Information av typen 'förbjudna områden i rymden' gör det i princip möjligt för robotsystemet att lägga ut en bana i rymden utan att riskera kollisioner med en känd omvärld. Att automatiskt planera rörelser med hänsyn tagen till objekt i robotsystemets omgivning ställer sig idag mycket svårt. Man blir ofta tvingad att göra förenklade antaganden om robotsystemets respektive objektens uppbyggnad. Kommandon, där flexibiliteten bör vara stor, återfinns främst inom sensor och återkopplingsfunktionerna, dels för att det är svårt att på förhand veta vilka applikationer och sensorer som ett system skall arbeta med och ta hand om, dels för att man kan vilja göra tillägg i styrsystemet i dess styralgoritmer och återkopplingsfunktioner.

Produktmodulen skall om symboliska namn används eller om man på annat sätt använder sig av referenser bakåt till en beskrivning av robotens världsbild leverera data relaterade till dessa referenser till robotens realtids styrsystem i explicit form. Explicita data för symbolnamn skall kunna editeras vilket innebär att sådan information bör vara lagrad i textform. Speciellt skall denna modul kunna utnyttja andra databaser, t ex CAD/CAM-databaser för att från dessa kunna beräkna erforderliga data som behövs för att utföra ett kommando som refererar till explicita data genom symboliska namn. Kommandot i exemplet ovan innebär att JOINT_1 och punkterna A och B ligger inlagda i ett CAD/CAM-system och att robotsystemet via produktmodulen har tillgång till denna information. Detta ger stora möjligheter att utforma robotarbetsplatser i ett CAD/CAM-system för att sedan utnyttja denna kunskap (datamängd) vid programmeringen av roboten. Speciellt ser man stora möjligheter att utnyttja CAD/CAM-system vid programmeringen av roboten som fallet är vid programmeringen av NC-maskiner. Man kan dock inte göra en direkt jämförelse då en robot vid en arbetsuppgift eller process har betydligt fler frihetsgrader i logik, sekvens och processutförande än vad en NC-maskin har beroende på den mångsidighet som en robot har kontra NC-maskinens vilken arbetar inom ett mycket smalt och specialiserat område. För kommandon av formen OPTIMIZE WELDPARAMETERS tillkommer snitt mot expertsystem som ur givna indata skall söka optimera svetsparametrarna för den givna operationen.

Realtids styrsystemmodulen slutligen utför själva styrningen av robotsvetsstationen. Styrsystemmodulen erhåller kommandon och styrdata från programmodulen samt numerisk och annan tilläggsinformation från produktmodulen om symboliska namn och referenser till en kunskapsdatabank har använts i applikationsprogrammet. En variant på informationsutbytet kan vara att programmodulen samlar in alla erforderliga data från produktmodulen som behövs vid exekveringen av ett applikationsprogram för att sedan lägga dessa tillsammans med applikationsprogrammet i en gemensam fil som realtids styrsystemet sedan har tillgång till under exekveringen.

I fortsättningen kommer endast uppbyggnaden av realtids styrsystemmodulen samt vissa svetsrelaterade delar i produktmodulen att behandlas, då det är denna som bestämmer ett robotsystems möjligheter att fungera i den miljö som

förekommer vid flexibel automatiserad svetsning. För systemets totala effektivitet är det däremot viktigt att samtliga huvudmoduler som är beskrivna ovan understödjer varandra på samma automations- och informationsnivå, eftersom stora rationaliseringsvinster kan göras om produktionsberedningen, programmeringen och inkörningen av en svetsrobotstation kan göras snabbare och effektivare.

5.1 REALTIDS STYRSYSTEM

För att uppnå ett sensorinteraktivt målsökande styrsystem enligt de angivna förutsättningarna krävs tre sinsemellan samarbetande undersystem, se bild 5.2:

- Rörelsesystem
- Sensorsystem
- Målgenerator

Rörelsesystemets uppgift är att succesivt bryta ner kommandon från applikationsprogrammet till explicita rörelsebeskrivningar och fysiska aktiviteter. Tekniken kan principiellt beskrivas enligt bild 4.12 i föregående kapitel, där ett arbetsbeskrivande kommando bryts ner till en rad delmoment. Dessa bryts sedan i sin tur ner till delmoment, osv till dess att man kan generera de fysiska rörelserna hos manipulatorn.

Sensorsystemet samlar in och processar data från de sensorer som systemet utnyttjar vid en given arbetsuppgift. Det resultat som denna processering skall ge skall vara av ett i systemet specificerat och normaliserat format. Den information som skall processas skall också kunna styras från applikationsprogrammet. Den information som sensorsystemet hämtar in och processar kan vara

- aktuell hastighet
- aktuell position i rummet för en arbetspunkt
- avvikelser i rummet för en arbetspunkt från avsedd bana
- detektion av föremål

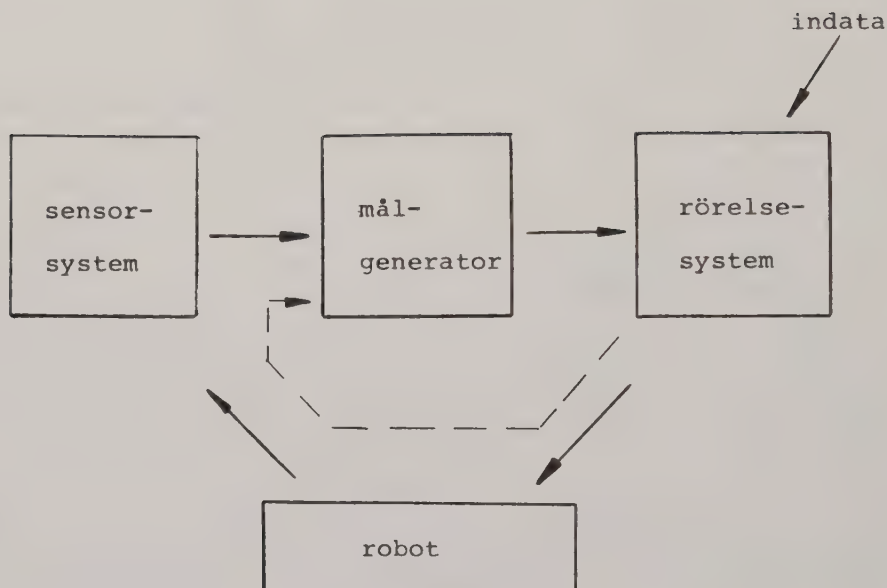


Bild 5.2 Blockschemat över ett reelltids styrsystems huvudfunktioner [5.1].

- speciell extraherad information från visionsystem
- processövervakning
- haveriövervakning

Målgeneratoren innehåller återkopplingsalgoritmer för manipulatorns sensor-interaktiva beteende och bestämmer hur informationen från sensorsystemet skall användas. Målgeneratoren skall därför ha möjlighet att ändra manipulatorns rörelsesätt under pågående aktivitet (t ex följning vid bågsvetsning) men även som tidigare antytts kunna stoppa pågående aktivitet då arbetsförloppet börjar urarta och söka nya angreppsvägar för att fullfölja arbetsuppgiften. För att veta om processen fortlöper normalt eller ej krävs funktioner i målgeneratoren som utvärderar och bedömer det arbete som robot-systemet utför samt funktioner som kan ange ett alternativt handlingssätt eller återkopplingsstrategi då processen urartar.

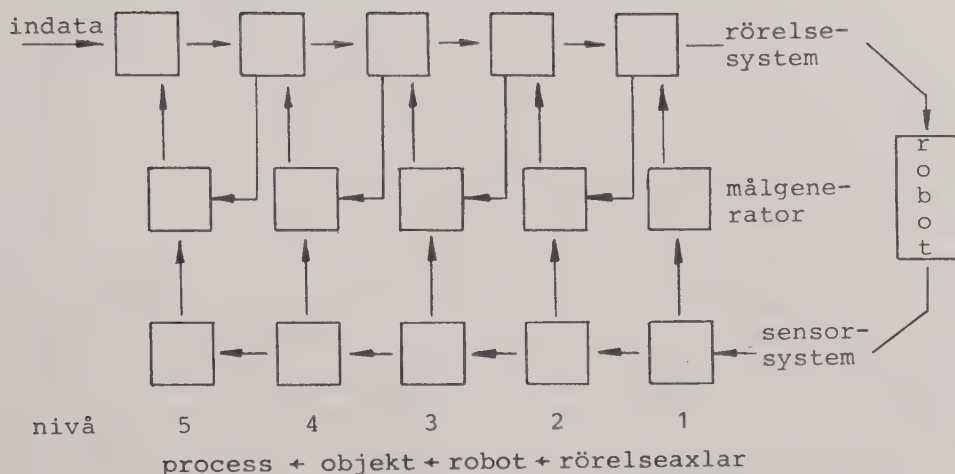


Bild 5.3 Blockschema över ett hierarkiskt realtids styrsystem [5.1].

5.1.1 HIERARKISK UPPBYGGNAD AV STYRSYSTEMET

För att kunna utföra komplexa kommandon av arbetsbeskrivande art, agera som strateg vid val av alternativa handlingssätt samt processa sensordata av skiftande komplexitet måste varje huvudfunktion i realtids styrsystemet byggas upp som en hierarki, se bild 5.3. Detta gör det möjligt att bygga upp ett beskrivet system underifrån med funktioner av stigande abstraktion utan att systemets struktur ändras eller att tidigare arbete måste göras om.

Hur sker den stegvisa förfiningen av en arbetsbeskrivning i det hierarkiska systemet? Om man som exempel tar ett indatakommando av typen

WELD JOINT_1 FROM A TO B, OPTIMIZE WELDPARAMETERS

krävs det en mängd beslutsval och kunskap vid nedbrytningen av kommandot till fysiska rörelser. För en mängd standardfall (som t ex kommandot ovan) skall målgeneratorn innehålla en rad standardstrategier med olika prioritet för

variationer i den aktuella processen. Detta kan vara hur robotens kända omvärld ser ut, hur roboten skall närma sig svetsfogen med hänsyn till robotsvetsstationens konfiguration och fixturens och objektets utformning, position och orientering, etc. Under rörelsen skall dessutom målgeneratören kunna aktivera lämpliga sensorsystem och använda sig av lämpliga återkopplingsalgoritmer samt ange vilken prioritet dessa skall få i rörelsesystemet för att styra manipulatern under pågående process till måluppfyllelse.

Om man studerar bild 5.3, kan man se att systemet består av tre hierarkier (rörelsesystem, målgenerator och sensorsystem) som arbetar i fem nivåer med ökande abstraktionsgrad. Systemets arbete i realtid administreras och tidsplaneras av en scheduler. Kommunikationen med yttervärlden (andra databaser, manipulatern, sensorsystem) sköts av en separat I/O-rutin. Abstraktionsgraden i systemet följer sedan följande schema nedifrån och upp:

Nivå 1: Arbetar på robotnivå med utdata direkt till manipulaterns styrenheter på rörelseaxlarna. Utdataformatet är av typen digitala pulser i varje klockcykel (typiskt var 10:e ms), där antalet pulser som ges motsvarar den sträcka som manipulaterns axlar skall tillryggalägga under en klockcykel. Indata är manipulaterns positioner (för rörelser) i robotens koordinatsystem (ledvinklar för revoluta robottyper).

Nivå 2: Arbetar i robotkoordinater. Detta innebär att man vid t ex en robot med revoluta koordinater representerar robotens rörelser med robotens ledvinklar. Utdata (positionen i robotkoordinater) är indata till nivå 1 och robotens positioner i ett kartesiskt koordinatsystem är indata till denna nivå.

Nivå 3: Arbetar med enkla rörelser (linjära, cirkulära, vektor, splines, etc) i ett kartesiskt koordinatsystem. På denna nivå sker all interpolering av manipulaterns rörelser. Utdata till nivå 2 blir manipulaterns position i ett kartesiskt koordinatsystem i nästa klockcykel. Indata till nivån är de elementära rörelsernas start- och slutpunkter samt

andra nödvändiga data (t ex radie för cirkel).

Nivå 4: Arbetar med enkla sammansatta rörelser som kan vara fördefinierade (och eventuellt alternativa) makron bestående av rörelser och logiska sekvenser för utförandet av enklare uppgifter som t ex 'WELD JOINT_1 ...'. Utdata till nivå 3 blir en serie rörelser och logiska sekvenser som söker utföra den avsedda uppgiften. Indata till nivån består av en samling kommandon från nivå 5 som är sammansatta till macron. Dessa kan vara fördefinierade 'manuellt' eller genererade automatiskt genom automatisk processplanering.

Nivå 5: Arbetar med målbeskrivande arbetsuppgifter som t ex 'WELD OBJECT A'. Utdata till nivå 4 är enklare delmoment i arbetsuppgiften som kan beskrivas med ett fåtal enklare rörelse- eller logikscheman. Indata till nivån är kommandon av målbeskrivande typ. För att kunna bryta ner ett sådant kommando till aktiviteter av fysisk art krävs som tidigare nämnts ofta att robotsystemet har tilläggsinformation som beskriver hur robotens omvärld ser ut samt att systemet har rutiner som kan välja ut en strategi på ett effektivt sätt och testa den i sin fysiska miljö.

I denna hierarki arbetar målgeneratoren och sensorsystemet på likvärdig nivå med rörelsesystemet så att återkopplingsstrategierna och sensordata matchar den information och de kommandon som skall genereras och utföras.

Med beaktande av vad som tidigare har framkommit under kravspecifikation, systemstruktur och sensorer ges här en beskrivning av hur de lägre nivåerna samt dessas inbördes struktur kan konstrueras för att bilda ett sensorinteraktivt styrsystem.

Man kan konstatera enligt tidigare att så länge som en industrirobot används i en relativt stel automation (vilket ofta är fallet idag), där antalet sensorer eller brytpunkter i robotens funktionssätt är mycket begränsade, är uppbyggnaden eller strukturen på styrsystemet av mindre betydelse. Man vet på förhand hur ett kommando eller en kommandosekvens skall utföras utan att

behöva bekymra sig om avbrott eller modifieringar från sensorer som talar om att något har hänt i omgivningen vilket bör leda till någon form av åtgärd (undantaget nödstoppsrutiner som krävs i alla robotsystem).

I och med införandet av sensorinteraktivitet med omgivningen med många möjliga brytpunkter eller valmöjligheter i utförandet av en uppgift, kommer komplexiteten på styrproblemet att öka dramatiskt och det blir i praktiken omöjligt att skriva styrsystemets mjukvara som ett enda program. Ett normalt tillvägagångssätt vid lösandet av komplicerade uppgifter är att använda sig av en modulariserad uppbyggnad.

För att fungera i ett beskrivet robotsystem kommer modulerna att ges egenskaper så att de var för sig fungerar som självständiga processer. Modulerna skall vid varje klockpuls läsa sina indata, beräkna utdata och skriva ut dessa till respektive moduls utgång. Om modulerna har dessa egenskaper, kommer systemet att kunna samarbeta med sin omgivning (ta emot sensordata) på varje nivå i hierarkin, beräkna och generera mål som uppfyller den aktuella uppgiften i realtid enligt den tidsdelning och krav som ställs på systemet. En ytterligare fördel med en uppdelning av modulerna som självständiga processer är att det ger stora möjligheter för uppbyggnad av ett beskrivet system med parallell processering, där funktionerna i varje modul beräknas av var sin dator (eller processor) och att datorerna i sin tur är kopplade till varandra i ett nätverk för dataöverföring till andra moduler och databaser.

Vissa grundstenar i ett beskrivet sensorinteraktivt styrsystem har studerats, byggts upp och testats. I följande avsnitt skall dessa delar beskrivas mera i detalj var för sig.

5.2 FÖRSÖKSUTRUSTNING FÖR EXPERIMENTELLA STUDIER

Den experimentella utrustningen som använts i arbetet har utnyttjats för att simulera och testa intressanta arbetssätt och metoder vid uppbyggnaden av ett robotsystem för småserietillverkning vid bågsvetsning. Huvuddelarna i experimentutrustningen består av:

- Industrirobot ASEA IRb6/S1 med datorlänk QHNM 318V för seriell datakommunikation med extern dator
- Svetsströmkälla ESAB LAH630 med trådmatarverk A10-MEC44 och programenhet A6-PA 16 för förinställning av svetsparametrar
- Dator VAX/VMS 11/780 för beräkning av styrdata och simuleringar
- Dator JET-80 med expansionsenhet för ECB-buss för kommunikation mellan VAX-dator och robot
- Terminal Tektronix 4105 (grafisk) för simulering och kontroll av styrdata samt Tidata (alfanumerisk) för programskrivning

Industriroboten IRb6/S1 är en generell robot med fem frihetsgrader avsedd att användas för bl a processuppgifter som bågsvetsning. Lastkapaciteten hos robotarmen är sex kg vilket är tillräckligt för de krafter som uppstår vid bågsvetsning. Styrsystemet arbetar enligt PTP-styrningsprincipen med möjlighet att utföra en synkroniserad interpolering av robotaxlarna i robotkoordinater. Svetsströmkällan är ansluten så att robotens styrsystem kan styra strömkällans funktioner och parametrar. Genom programenheten A6-PAF 16 kan fem olika förvalda svetsparametrar väljas från styrsystemet. Med hjälp av datorlänken kan man upprätta kommunikation med en överordnad dator. Länken utnyttjas för att sända inkrementella kommandon eller överföra program som t ex är genererade med ett off line programmeringsspråk.

VAX-datorn används för uppbyggnad av vissa delar av ett styrsystem för robotsvetsstationen samt styrning av roboten. För att få en snabbare överföring till roboten än vad datorlänken klarar av skickas datainformation via seriell ledning med 19200 baud till JET-datorn som via ett inklockningsförfarande skickar styrinformationen vidare till de enskilda styrkortet för robotens rörelseaxlar. På detta sätt har endast robotens styrsystems styrkort med servoenhet och drivdon utnyttjats i de praktiska försöken.

5.3 GENERERING AV UTDATA

Processeringen av styrdata som skall generera fysiska aktiviteter på roboten kan beskrivas som den schematiska framställningen enligt bild 5.4. Närmast roboten i hierarkin finns en servoenhet (en per rörelseaxel) som ser till att robotaxlarna styr in sig till den position som är definierad som ett börvärde. På varje rörelseaxel finns det därför en sensor eller mätton som mäter aktuell position (ärvärdet). Om man jämför ärvärdet och börvärdet, får man ett mått på hur långt en rörelseaxel skall röra sig för att komma i rätt läge (ärvärdet blir lika med eller ligger mycket nära börvärdet). Indata (börvärdena) till servoenheten är den information som anger hur roboten skall styras ut till olika rörelser och bildar alltså slutprodukten av det sensorinteraktiva styrsystemet.

Den lägsta nivån (nivå 1) genererar en serie pulser eller binära tal till varje servostyrd rörelseaxel. Systemet arbetar med en fast tidbas på 10 ms [5.2]. Servoenhetens indata under en klockcykel skall därför motsvara den sträcka som en rörelseaxel skall tillryggalägga på 10 ms. Detta innebär att utdata från nivå 1 till servoenheten skall vara digitala heltalsvärden som betecknar hastigheter på rörelseaxlarna räknat som robotinkrement per tidsenhet. Med robotinkrement menas här den minsta upplösbara förflyttning som robotens styrsystem kan styra ut. Inkrementen definieras utifrån mätsystemet på drivdonen, elektroniken i servoenheten och hur överföringen sker av en rörelse från drivdonen via transmissioner ut till rörelseaxlarna.

Indata till nivå 1 i hierarkin är den rörelse som roboten skall utföra i termer av vridningar eller i förekommande fall av linjära förflyttningar av manipulatorens armar räknade i radianer eller millimeter. Dessa indata beskriver robotens rörelse i robotkoordinater.

Uppgiften för den lägsta nivån i hierarkin blir därför att utifrån de önskade ledpositionerna på manipulatorens armar beräkna hur många robotinkrement som varje rörelseaxel skall förflyttas för att komma till önskad position. Beräkningarna i nivå 1 utförs enligt den matematik som ges i avsnitt 5.3.1 nedan.

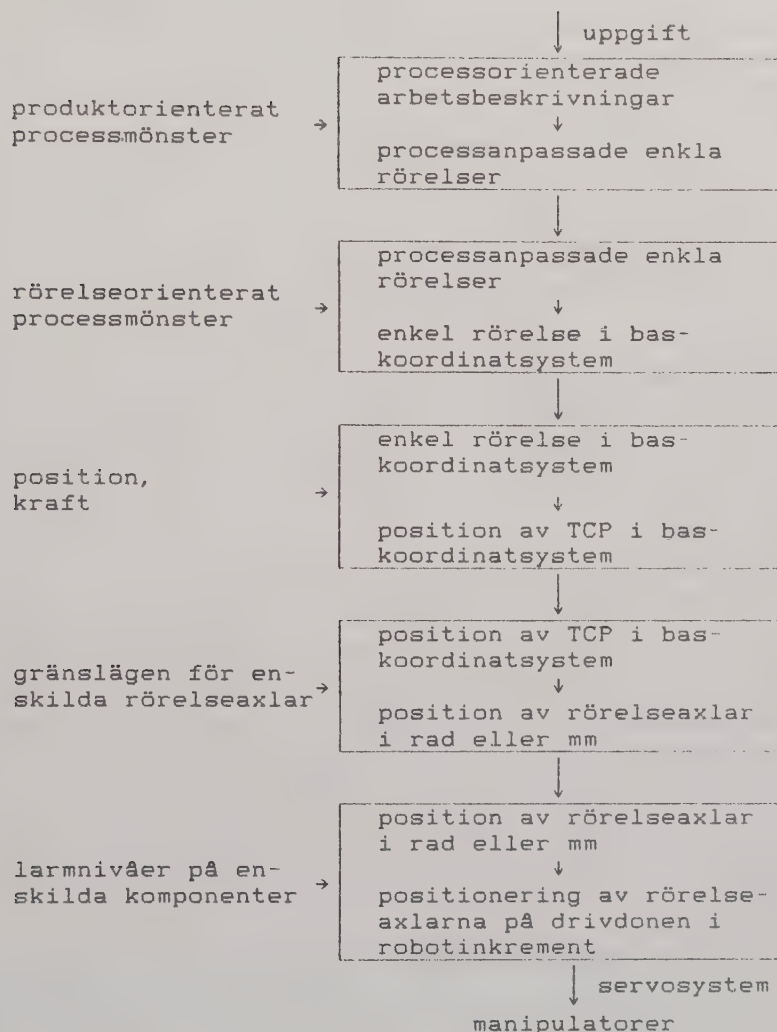


Bild 5.4 Schematisk skiss över den rörelsegenererande hierarkin med exempel på återkopplingsdata från sensorsystemet. Ett kommando i toppen som beskriver en enkel uppgift kommer succesivt att brytas ner till rörelser av de enskilda rörelseaxlarna på manipulatern.

Återkopplingen från sensorsystemet begränsas som regel på denna nivå till att gälla olika former av nödstopp i systemet som t ex avkänning av differensen

mellan bör- och ärvärde, kraftpåkänning på enskilda drivdon eller begräning av hastigheten vid utstyrningen av robotinkrement till de olika rörelseaxlarna under varje tidsinkrement.

Den andra nivåns uppgift i hierarkin blir att generera rörelsedata i robotens koordinatsystem till den första nivån. Detta innebär att utdata från den andra nivån är de positioner i robotkoordinater som robotens rörelseaxlar skall ställa sig i, vilket är lika med den första nivåns indata. De beräkningar som utförs i den andra nivån är huvudsakligen att transformera en rörelse av roboten beskriven i ett kartesiskt koordinatsystem till en rörelse av roboten beskriven i robotens koordinatsystem. Detta betyder att indata till nivå 2 är robotens arbetspunkt beskriven i ett kartesiskt koordinatsystem (xyz samt orientering). Beräkningarna i nivå 2 utförs enligt den matematik som ges i avsnitt 5.3.2 nedan.

Aterkopplingen från sensorsystemet är även här begränsat till axelspecifika data. På denna nivå gäller det främst bevakning av gränslägen för enskilda rörelseaxlar. Dessa kan vara verkliga eller beteckna förbjudna områden för aktuell robotstation som t ex begränsning av robotkroppens vridning då roboten arbetar mot ett objekt. Detta möjliggör en säker arbetsplats vid en andra arbetsstation inom robotens arbetsområde för förberedelser av ett annat objekt.

Den tredje nivåns uppgift i hierarkin är att generera utdata till den andra nivån i form av en serie förflyttningar beskrivna i ett kartesiskt koordinatsystem som stämmer överens med den önskade rörelsebanan. De rörelser som den tredje nivån genererar kan beskrivas som 'enkla rörelser' i rymden, t ex linjär eller cirkulär. Den tredje nivåns huvuduppgift är alltså interpolering av robotens rörelser i ett kartesiskt koordinatsystem, dvs en beräkning av hur robotens arbetspunkt skall genomlöpa en rörelsebana i små inkrementella delsträckor för att generera önskad rörelse med tillräcklig noggrannhet. Positionsdata för varje delsträcka är utdata från nivå 3 och blir indata i nivå 2, där man utifrån denna information beräknar aktuell position uttryckt i robotkoordinater. Indata till nivå 3 är kommandon som beskriver robotens

rörelse i ett kartesiskt koordinatsystem genom enkla rörelser. Dessa rörelser kan som tidigare nämnts vara av typen linjär eller cirkulär med specificerad hastighet. För bågsvetsning gäller dessutom att rörelser som kan beskrivas med en vektor som i realtid kan ändras under varje tick av en fogföljarsensor är av mycket stort värde för att möjliggöra en fri fogföljning under svetsning. Beräkningarna i nivå 3 utförs enligt den matematik som ges i avsnitt 5.3.3 nedan.

De återkopplingar som sker från denna nivå är av en mer avancerad form och påverkar robotarmens rörelse som en integrerad helhet sett utifrån ett kartesiskt baskoordinatsystem. Typ av information som återkopplas på denna nivå är rörelser (fogföljning), krafter (montering, slipning) och positionsstyrning inom begränsade volymer (plocka detaljer med hjälp av information från ett visionsystem).

I och med att man kommer upp till nivå 4 i hierarkin börjar man lämna den i realtid direkta styrningen av robotens rörelser. Ett kommando på nivå 3 beräknas 'kontinuerligt' under interpoleringen i nivå 3, 2 och 1 med en frekvens på ca 10 ms, medan styraktiviteter på nivå 4 har en frekvens som räknas i sekunder eller däröver. De styraktiviteter som utförs av nivå 4 är mera av process- eller objektbeskrivande art än rörelsebeskrivande. Detta medför att man måste ha god kännedom och ha goda kunskaper och erfarenheter om de processer som man vill styra. Exempel på sådana styrfunktioner kan vara

- logik- och rörelseschema för svetsstart och svetslut
- optimering av samspelet svetsrobot och lägesställare
- sökning av detaljer inom en volym
- strategi för rörelsebana med ingående hinder
- optimering av svetsparametrar

Ett matematiskt underlag för sökning av samordnade rörelser mellan lägesställare och robot med hänsyn tagen till processen ges i avsnitt 5.3.4 nedan.

De sensordata som nivå 4 skall återkopplas med blir av mera kvalificerad art än tidigare. Man har t ex vision för att utföra säkra banrörelser i rymden samt identifiera föremål, närhetssensorer för att utföra sökning i en volym, speciella kraftsensordata för att utföra passningsoperationer samt process-orienterade sensorer för att optimera processparametrar etc. I och med detta ser man att kommandospråket till systemet på högre nivåer kommer att bli mera abstrakt och vara av mer arbets- eller processuppfyllande art.

Nivå 5 arbetar på processnivå med arbetsbeskrivande instruktioner. Genom kunskap om objektet och vilka fogar som skall svetsas skall en svetsföljd samt mellanliggande rörelser bestämmas. Den information som skall genereras till nivå 4 är de enskilda fogarnas positioner, orienteringar och beräknade processdata under svetsning. Den processkunskap som krävs för att ge sådan information är svår att ange explicit. En anledning till detta är problemets svårighetsgrad och stora antal parametrar (t ex fogtyper, material, dimensioner), vars relationer inte tillfullo är helt kända.

Ett system med högre funktioner kommer att ta tid och erfarenhet från praktiska applikationsfall att definiera fullständigt, om man vill kunna ta hand om ett stort antal olika typer av arbetsuppgifter eller processer. Det är därför viktigt att dessa moduler och styrfunktioner ges en sådan utformning att det är enkelt att tillföra nya funktioner eller kommandon till en modul samtidigt som in- och utdata skall kunna hållas konstanta.

Vid beräkning av styrdata till roboten kommer styrrutinerna i nivå 1, 2 och 3 att kräva snabba algoritmer, då utdata skall genereras på i storleksordningen var 10:e ms. I nivå 1 är det som regel ett trivialt problem att nå en effektiv överföringsfunktion, då förhållandet mellan robotkoordinater och robotinkrement oftast är enkelt att lösa (icke linjära överföringsfunktioner kan dock förekomma som vid över- och underarmen på ASEA IRb6).

Transformeringen mellan robotkoordinater och kartesiska koordinater som förekommer i nivå 2 är av en helt annan arbetsinsats räknat i antalet matematiska operationer som måste utföras vid transformeringen mellan koordinat-

systemen. Man har två transformeringar att beräkna dels robotens arbetspunkt utifrån kända ledpositioner (det direkta kinematiska problemet), dels vilka ledpositioner som roboten skall stå i för att uppnå en önskad position och orientering med en given arbetspunkt på roboten (det inversa kinematiska problemet) enligt diskussionen ovan i kapitel 4.1.1.

För 6-axliga robotar med tre skärande handledsaxlar finns en explicit lösning på det inversa problemet. Problemet vid styrning av en 5-axlig robot är att man saknar möjlighet att positionera robotens arbetspunkt till ett godtyckligt läge och orientering i robotens arbetsområde. Detta gör att man i nivå 3 får definiera orienteringen av arbetspunkten i robotens handledskoordinater och flytta upp orienteringen av arbetspunkten i ett kartesiskt koordinatsystem till en optimerings- och styrfunktion i nivå 4 relaterat till den process som skall utföras. Detta är speciellt viktigt vid processuppgifter som bågsvetsning, där svetspistolens orientering mot arbetsstycket är av stor betydelse för resultatet. Även för 6-axliga robotar finns det i praktiken stora områden som inte är möjliga att uppnå med godtycklig orientering på grund av gränslägen för de olika rörelseaxlarna. För robotsvetsstationer med flera än sex rörelseaxlar uppstår redundans i det kinematiska systemet genom att man får ett överbestämt styrproblem. Detta gör att det krävs optimerings- och kontrollrutiner som styr och beräknar hur svetspistolen och objekt skall förhålla sig till varandra för att nå ett för processen gynnsamt resultat.

5.3.1 NIVÅ 1 - GENERERING AV STYRDATA TILL RÖRELSEAXLARNAS

Den lägsta nivån i rörelsehierarkin består av en funktion som direkt överför en styrintegration till varje rörelseaxels styr- och servoenhet för att generera en fysisk rörelse av respektive rörelseaxel. För att erhålla en fysisk rörelse hos roboten skall man var 10:e ms ange en ny position för de olika rörelseaxlarna i robotinkrement. Den nya positionen skall anges relativt aktuell position som ett tal med sju bitar samt en teckenbit (+/- 127). Överföringen av information sker genom ett inklockningsförfarande [5.2].

Indata till nivå 1 är en beskrivning av robotaxlarnas positioner räknat i vridningsvinklar (radianer). Varje rörelseaxels ledposition skall sedan över-sättas till ett koordinatsystem som bestäms utifrån enheten robotinkrement.

Robotinkrementen betraktas internt i robotens styrsystem som 16 bitars positiva tal. Utgående från definitionen av vinklarna och dessas nolläge kommer respektive robotaxel att ha ett positivt inkrementellt koordinatvärde. Man kommer därför att för varje överföringsfunktion från vridningsvinkel till robotinkrement att få en offset i robotinkrement som adderas till den inkrementella vridningen ifrån dess nolläge. Nedan följer en beskrivning av hur denna överföring sker uppdelat på de olika rörelseaxlarna ϕ (kroppsrotation), θ (underarm), α (överarm), τ (handledsböjning) och ψ (handledsrollning).

5.3.1.1 KROPPSROTATIONEN ϕ

För ϕ -axeln gäller ett linjärt förhållande mellan vridningsvinkeln och antalet robotinkrement som krävs att utföra vridningen. För detta förhållande gäller att en vridning av rörelseaxeln med 1 radian motsvarar 5000 robotinkrement och man erhåller följande samband:

$$\phi_{inc} = (\phi * \phi_{gear}) + \phi_{incoff} \quad (5-1)$$

där ϕ_{inc} är den inkrementella koordinaten för ϕ -axeln, ϕ är ϕ -axelns vridning i radianer, ϕ_{gear} är utväxlingsförhållandet (5000 inc/radian) och ϕ_{incoff} är inkrementell offset då ϕ -axeln befinner sig i mittläget ($\phi=0$).

För det omvända fallet, där den inkrementella koordinaten ϕ_{inc} är känd får man ur (5-1) att

$$\phi = \frac{\phi_{inc} - \phi_{incoff}}{\phi_{gear}} \quad (5-2)$$

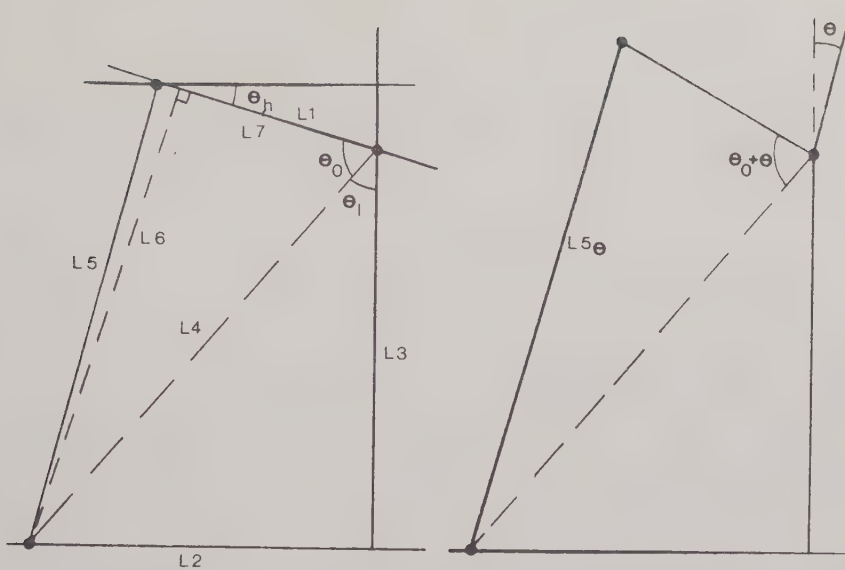


Bild 5.5 Länksystem som genererar θ -rörelsen.

5.3.1.2 UNDERARMEN θ

För θ -vinkeln uppstår ett olinjärt förhållande mellan vridningsvinkel och antal robotinkrement beroende på det länksystem som rörelsen uppstår ur, se bild 5.5.

I länksystemet är $L1$, $L2$, $L3$ och θ_h givet, där $L1$ är hävarmslängden mellan övre leden för kulmutterskruven och nedre leden för underarmen, $L2$ är avståndet från basen ut till ledfästet för kulmutterskruven, $L3$ är avståndet från basen upp till leden för underarmen och θ_h är vinkeln mellan horisontallinjen och hävarmen $L1$. Man erhåller följande samband:

Längden $L4$ erhålles som

$$L_4 = \sqrt{L_2^2 + L_3^2} \quad (5-3)$$

vinkeln θ_1 erhålles som

$$\theta_1 = \arctan \frac{L_2}{L_3} \quad (5-4)$$

Vinkeln θ_0 erhålles för $\theta=0$ som

$$\theta_0 = \frac{\pi}{2} - \theta_1 + \theta_h \quad (5-5)$$

Ur dessa samband kan längden L_5 för kulmutterskruven erhållas ur cosinussatsen som

$$L_5 = \sqrt{L_1^2 + L_4^2 - 2 * L_1 * L_4 * \cos(\theta_0)} \quad (5-6)$$

eller då vridningsvinkeln (θ) för underarmen varierar enligt

$$L_{5\theta} = \sqrt{L_1^2 + L_4^2 - 2 * L_1 * L_4 * \cos(\theta_0 + \theta)} \quad (5-7)$$

Den längdförändring som uppstår på kulmutterskruven kan skrivas som

$$L_\delta = L_{5\theta} - L_5 \quad (5-8)$$

För kulmutterskruven gäller ett utväxlingsförhållande att en vridning av ett varv ger en längdförskjutning av 5 mm. Detta motsvarar också 200 robotinkrement. Man har alltså ett utväxlingsförhållande på kulmutterskruven så att 1 mm längdförändring motsvarar 200/5 robotinkrement. Med ledning av ovanstående samband erhålles koordinatvärdet i robotinkrementen för θ -axeln som funktion av θ som

$$\theta_{inc} = L_{\delta} * \theta_{gear} + \theta_{incoff} \quad (5-9)$$

där θ_{inc} är den inkrementella koordinaten för θ -axeln, L_{δ} är längdförändringen av kulmutterskruven från mittläget ($\theta=0$), θ_{gear} är utväxlingsförhållandet och θ_{incoff} är inkrementell offset då θ -axeln befinner sig i mittläget.

För det omvända fallet, där den inkrementella koordinaten θ_{inc} är känd, kan man ur (5-8) och (5-9) beräkna $L5_{\theta}$ som funktion av θ enligt

$$L5_{\theta} = \frac{\theta_{inc} - \theta_{incoff}}{\theta_{gear}} + L5 \quad (5-10)$$

Om vi sätter $s = (L1 + L4 + L5_{\theta})/2$, erhålles arean A för triangeln $L1$ - $L4$ - $L5_{\theta}$ enligt Heron's formel

$$A = \sqrt{s(s-L1)(s-L4)(s-L5_{\theta})} \quad (5-11)$$

och höjden $L6$ på triangeln $L1$ - $L4$ - $L5_{\theta}$ som

$$L6 = \frac{2A}{L1} \quad (5-12)$$

Längden L_7 erhålles som

$$L_7 = \sqrt{L_4^2 - L_6^2} \quad (5-13)$$

Ur (5-12) och (5-13) kan slutligen vinkeln θ beräknas som

$$\theta = \arctan \frac{L_6}{L_7} - \theta_0, \quad 0 < \theta < \pi/2 \quad (5-14)$$

5.3.1.3 ÖVERARMEN α

För α -vinkeln uppstår ett olinjärt förhållande mellan vridningsvinkel och antal robotinkrement beroende på det länksystem som rörelsen uppstår ur, se bild 5.6.

I länksystemet är L_1 , L_2 , L_3 givet, där L_1 är hävarmslängden, L_2 är avståndet från basen ut till ledfästet för kulmutterskruven och L_3 är avståndet från basen upp till leden för θ -axeln. En rörelse av hävarmen L_1 leder genom länkmekanismen till att motsvarande rörelse genereras på α -axeln. Man erhåller följande samband:

Längden L_4 erhålles som

$$L_4 = \sqrt{L_2^2 + L_3^2} \quad (5-15)$$

vinkeln α_1 erhålles som

$$\alpha_1 = \arctan \frac{L_2}{L_3} \quad (5-16)$$

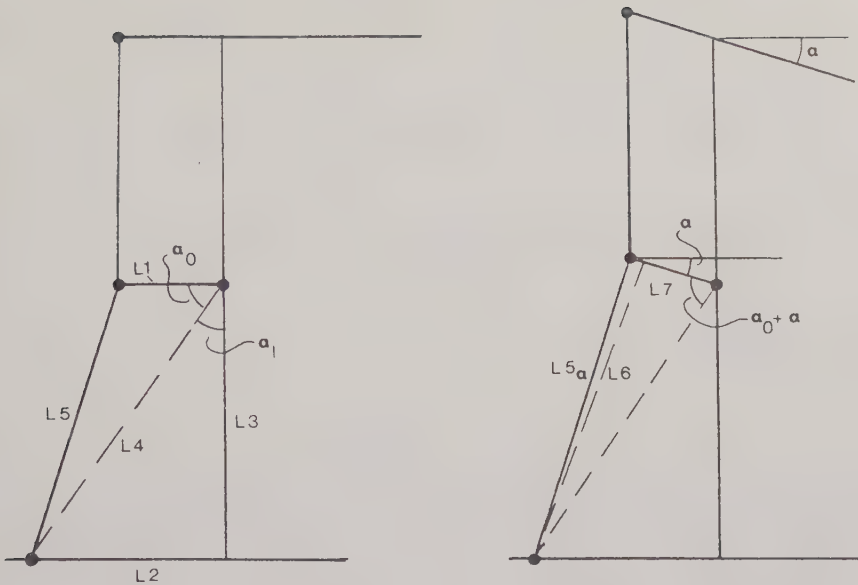


Bild 5.6 Länksystem som genererar α -rörelsen.

Vinkeln α_0 erhålles för $\alpha=0$ som

$$\alpha_0 = \frac{\pi}{2} - \alpha_1 \quad (5-17)$$

Ur dessa samband kan längden $L5$ för kulmutterskruven erhållas ur cosinussatsen som

$$L5 = \sqrt{L1^2 + L4^2 - 2 * L1 * L4 * \cos(\alpha_0)} \quad (5-18)$$

eller då vridningsvinkeln (α) för underarmen varierar enligt

$$L5_{\alpha} = \sqrt{L1^2 + L4^2 - 2 * L1 * L4 * \cos(\alpha_0 + \alpha)} \quad (5-19)$$

Den längdförändring som uppstår på kulmutterskruven kan skrivas som

$$L_{\delta} = L5_{\alpha} - L5 \quad (5-20)$$

För kulmutterskruven gäller ett utväxlingsförhållande att en vridning av ett varv ger en längdförskjutning av 5 mm. Detta motsvarar också 200 robotinkrement. Man har alltså ett utväxlingsförhållande på kulmutterskruven så att 1 mm längdförändring motsvarar 200/5 robotinkrement. Med ledning av ovanstående samband erhålles koordinatvärdet i robotinkrementen för α -axeln som funktion av α enligt

$$\alpha_{inc} = L_{\delta} * \alpha_{gear} + \alpha_{incoff} \quad (5-21)$$

där α_{inc} är den inkrementella koordinaten för α -axeln, L_{δ} är längdförändringen av kulmutterskruven från mittläget ($\alpha=0$), α_{gear} är utväxlingsförhållandet och α_{incoff} är inkrementell offset då α -axeln befinner sig i mittläget.

För det omvända fallet, där den inkrementella koordinaten α_{inc} är känd, kan man ur (5-20) och (5-21) beräkna $L5_{\alpha}$ som funktion av α enligt

$$L5_{\alpha} = \frac{\alpha_{inc} - \alpha_{incoff}}{\alpha_{gear}} + L5 \quad (5-22)$$

Om vi sätter $s = (L_1 + L_4 + L_5_\alpha)/2$, erhålles arean A för triangeln $L_1-L_4-L_5_\alpha$ enligt Heron's formel

$$A = \sqrt{s(s-L_1)(s-L_4)(s-L_5_\alpha)} \quad (5-23)$$

och höjden L_6 på triangeln $L_1-L_4-L_5_\alpha$ som

$$L_6 = \frac{2A}{L_1} \quad (5-24)$$

Längden L_7 erhålles som

$$L_7 = \sqrt{L_4^2 - L_6^2} \quad (5-25)$$

Ur (5-24) och (5-25) kan slutligen vinkeln α beräknas som

$$\alpha = \arctan \frac{L_6}{L_7} - \alpha_0, \quad 0 < \alpha < \pi/2 \quad (5-26)$$

5.3.1.4 HANDLEDSBÖJNINGEN τ

För τ -axeln gäller ett linjärt förhållande mellan vridningsvinkeln och antalet robotinkrement som krävs att utföra vridningen. För detta förhållande gäller, att en vridning av rörelseaxeln med 1 robotinkrement motsvarar 0.24 mrad. Man erhåller följande samband:

$$\tau_{inc} = (\tau * \tau_{gear}) + \tau_{incoff} \quad (5-27)$$

där τ_{inc} är den inkrementella koordinaten för τ -axeln, τ är τ -axelns vridning i radianer, τ_{gear} är utväxlingsförhållandet (1/0.00024 inc/radian) och τ_{incoff} är inkrementell offset då τ -axeln befinner sig i mittläget ($\tau=0$).

För det omvända fallet, där den inkrementella koordinaten τ_{inc} är känd, får man ur (5-27) att

$$\tau = \frac{\tau_{inc} - \tau_{incoff}}{\tau_{gear}} \quad (5-28)$$

5.3.1.5 HANDLEDSROLLNINGEN ϕ

För ϕ -axeln gäller ett linjärt förhållande mellan vridningsvinkeln och antalet robotinkrement som krävs att utföra vridningen. För detta förhållande gäller, att en vridning av rörelseaxeln med 1 robotinkrement motsvarar 0.41 mrad. För vridrörelsen gäller dessutom, att en rörelse av τ -axeln genererar en vridning av ϕ -axeln genom länkmssystemet för handleden. En önskad vridningsvinkel av ϕ -axeln vid samtidig rörelse av τ -axeln kräver därför kompensation av ϕ -axelns rörelse med avseende på storleken av τ -rörelsen. Man erhåller följande samband:

$$\phi_{inc} = (\phi * \phi_{gear}) + (\tau * \tau_{gear}) + \phi_{incoff} \quad (5-29)$$

där ϕ_{inc} är den inkrementella koordinaten för ϕ -axeln, ϕ är ϕ -axelns vridning i radianer, ϕ_{gear} är utväxlingsförhållandet (1/0.00041 inc/radian), τ och τ_{gear} som ovan för τ -axeln och ϕ_{incoff} är inkrementell offset då ϕ -axeln befinner sig i mittläget ($\phi=0$).

Det omvända fallet, där den inkrementella koordinaten ψ_{inc} är känd, får man ur (5-29) som

$$\psi = \frac{(\psi_{inc} - \psi_{incoff}) - \tau * \tau_{gear}}{\psi_{gear}} \quad (5-30)$$

eller med utnyttjande av (5-27) som

$$\psi = \frac{(\psi_{inc} - \psi_{incoff}) - (\tau_{inc} - \tau_{incoff})}{\psi_{gear}} \quad (5-31)$$

5.3.2 NIVÅ 2 - TRANSFORMERING MELLAN ROBOT- OCH KARTESISKA KOORDINATER

Transformeringen av en position beskriven i ett kartesiskt koordinatsystem till en beskrivning av positionen i robotkoordinater ligger i nivå 2 i rörelsehierarkin. Hur robotkoordinaterna är definierade, beror på robotens kinematik och hur rörelsen är definierad utifrån denna, se bild 5.7.

Indata till nivå 2 är en beskrivning av robotens arbetspunkt definierad i ett kartesiskt koordinatsystem med x, y, z samt orientering. För den aktuella roboten (5-axlig robot, ASEA IRb6/S1) är en lösning av det generella inversa kinematiska problemet inte möjlig enligt avsnitt 4.1.1. Man blir därför tvungen att explicit ange orienteringen som vridning i handledskoordinater. Utdata till nivå 1 i rörelsehierarkin blir den aktuella arbetspunkten beskriven i robotkoordinater. Denna transformering skall behandlas för en robot enligt bild 5.7.

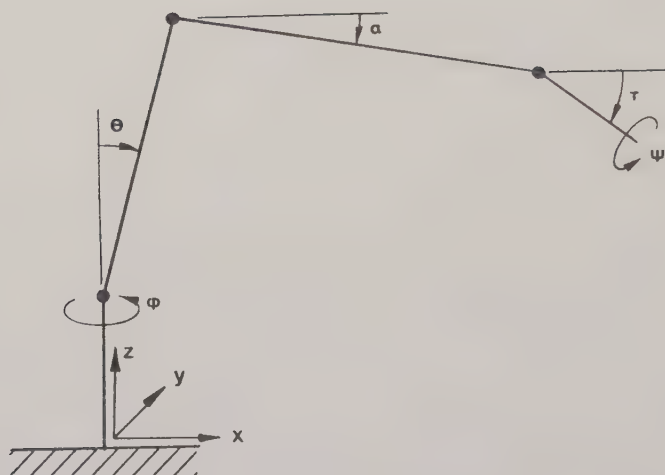


Bild 5.7 Robotarm med baskoordinatsystem och robotkoordinater.

5.3.2.1 DET DIREKTA KINEMATISKA PROBLEMET

Eftersom τ -axelns vridning definieras utifrån horisontallinjen kan man enkelt dela upp problemet i två mindre delar. Den första innebär en transformering från baskoordinatsystemet till centrum av τ -axeln. Den andra innebär en transformering från τ -axeln ut till arbetspunkten. För ϕ , θ och α -axlarna erhålles följande lösning av transformeringen från basen till handleden enligt bild 5.8.

Om positionen i handleden benämnes med index wrist, erhålles följande med benämningar enligt bild 5.8:

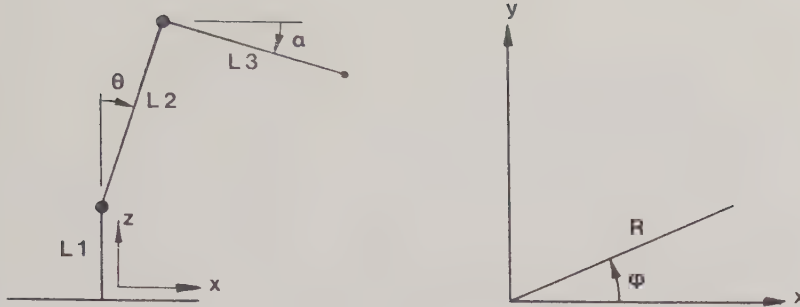


Bild 5.8 Definiering av φ , θ och α -axlarnas rörelser.

$$R_{\text{wrist}} = L2 * \sin \theta + L3 * \cos \alpha \quad (5-32)$$

$$X_{\text{wrist}} = R_{\text{wrist}} * \cos \varphi \quad (5-33)$$

$$Y_{\text{wrist}} = R_{\text{wrist}} * \sin \varphi \quad (5-34)$$

$$Z_{\text{wrist}} = L1 + L2 * \cos \theta - L3 * \sin \alpha \quad (5-35)$$

Den homogena transformationsformeln mellan baskoordinatsystemet och ett system $(X, Y, Z)'$ vridet runt z -axeln som φ -vridningen och translaterat till handleden med $(X, Y, Z)_{\text{wrist}}$ enligt ovan blir då

$$T_{\text{base}}^{\text{wrist}} = T_{\text{trans}} \cdot T_{\text{rotz}} \quad (5-36)$$

där

$$T_{\text{trans}} = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-37)$$

och

$$T_{\text{rot}, z} = \begin{bmatrix} \cos \zeta & -\sin \zeta & 0 & 0 \\ \sin \zeta & \cos \zeta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-38)$$

där dx , dy , dz är translationen från baskoordinatsystemet och ζ är vridningsvinkeln runt z -axeln. Genom att sätta in resultaten från (5-32) - (5-35) i (5-37) och (5-38) och beräkna T_{wrist} enligt (5-36) får man

$$T_{\text{base}}^{\text{wrist}} = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 & X_{\text{wrist}} \\ \sin \varphi & \cos \varphi & 0 & Y_{\text{wrist}} \\ 0 & 0 & 1 & Z_{\text{wrist}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-39)$$

För transformeringen från handleden ut till arbetspunkten erhålles en lösning med hjälp av homogena transformationsmatriser enligt följande, se bild 5.9:

$$T_{\text{TCP}}^{\text{wrist}} = T_{\text{rot}, \tau} \cdot T_{\text{trans}, L4} \cdot T_{\text{rot}, \phi} \cdot T_{\text{TCP}} \quad (5-40)$$

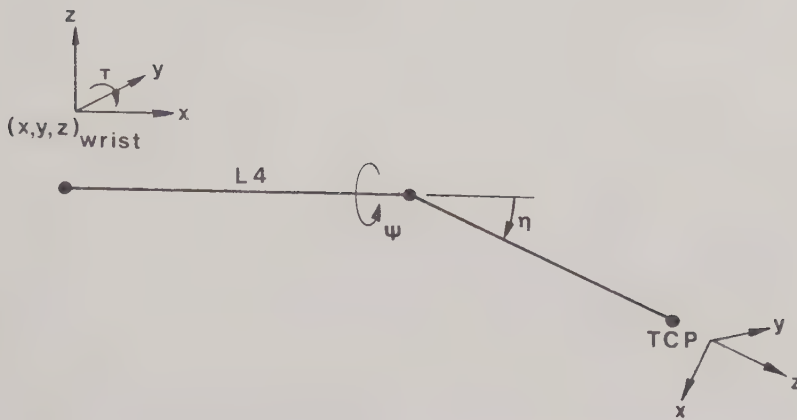


Bild 5.9 Handledssystemet för transformering mellan handleden och arbetspunkten.

och man får att

$$T_{TCP}^{wrist} = \begin{bmatrix} -\sin\tau \cos\phi & \sin\tau \sin\phi & \cos\tau & L4 \cdot \cos\tau \\ \sin\phi & \cos\phi & 0 & 0 \\ -\cos\tau \cos\phi & \cos\tau \sin\phi & -\sin\tau & -L4 \cdot \sin\tau \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot T_{TCP} \quad (5-41)$$

där T_{TCP} bestämmer hur arbetspunkten skall transformeras utifrån fästplattan på robotens ψ -axel. I det generella fallet har T_{TCP} formen

$$T_{TCP} = \begin{bmatrix} C\eta \ C\zeta & -C\eta \ S\zeta & S\eta & dx \\ S\zeta \ S\eta \ C\zeta & -S\zeta \ S\eta \ S\zeta & -S\zeta \ C\eta & dy \\ + \ C\zeta \ S\zeta & + \ C\zeta \ C\zeta & & \\ -C\zeta \ S\eta \ C\zeta & C\zeta \ S\eta \ S\zeta & C\zeta \ S\eta & dz \\ + \ S\zeta \ S\zeta & + \ S\zeta \ C\zeta & & \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-42)$$

där C , S anger cosinus- respektive sinusfunktionerna, ξ , η , ζ är rotationen av koordinatsystemet runt axlarna x , y , z och dx , dy , dz är avståndet från fästplattan på ψ -axeln till arbetspunkten för respektive koordinataxel. För en svetspistol gäller emellertid ett plant förhållande med vridning av koordinatsystemet runt en axel. Detta gör, att man endast behöver ta hänsyn till en av vinklarna ovan beroende på hur svetspistolen fästes på roboten. I praktiken kan man få göra en justering av koordinatsystemet i fästplattan beroende på om svetspistolen monteras vriden eller ej relativt ψ -axeln. Detta kan göras genom ytterliggare en transformation $T_{rot,z}$ innan transformationen T_{TCP} görs enligt (5-41) ovan eller genom att ändra ψ -axelns nollreferens. Det senare innebär även en förändring av gränserna för vridningen och inkrementell offset för ψ -axeln. Om denna justering utförs så, att orientering av koordinatsystemet kan göras kring y -axeln och med translation längs koordinatriktningarna ($dy=0$), får man T_{TCP} för en svetspistol som

$$T_{TCP} = \begin{bmatrix} \cos\eta & 0 & \sin\eta & dx \\ 0 & 1 & 0 & 0 \\ -\sin\eta & 0 & \cos\eta & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-43)$$

Med hjälp av (5-39), (5-41) och (5-43) kan transformationsmatrisen som beskriver transformationen från basen på roboten till TCP skrivas med matrismultiplikation som

$$T_{\text{base}}^{\text{TCP}} = T_{\text{base}}^{\text{wrist}} \cdot T_{\text{wrist}} \cdot T_{\text{TCP}} \quad (5-44)$$

där T_{wrist} anger transformationen från handledens τ -axel till fästplattan på ψ -axeln.

Resultatet av transformationsmatrisen i (5-44) ger verktygets orientering och position kolonnvis givet robotens ledpositioner så att

$$T_{\text{base}}^{\text{TCP}} = \begin{bmatrix} \xi_x & \eta_x & \zeta_x & p_x \\ \xi_y & \eta_y & \zeta_y & p_y \\ \xi_z & \eta_z & \zeta_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \xi & \eta & \zeta & p \end{bmatrix} \quad (5-45)$$

där ξ , η , ζ svarar mot verktygets x , y , z -riktningar enligt bild 5.9 relaterat till baskoordinatsystemet och p -vektorn anger verktygets position i baskoordinatsystemet.

5.3.2.2 DET INVERSA KINEMATISKA PROBLEMET

Det omvända fallet innebär att man utifrån en känd position för TCP samt orientering av verktyget beräknar, hur robotens leder skall positioneras. Enligt tidigare är det inte möjligt att erhålla en generell lösning av det inversa kinematiska problemet. Man blir därför tvungen att definiera TCP som positionen i ett kartesiskt baskoordinatsystem och orienteringen utifrån

handledskoordinater, dvs med τ - och ϕ -axlarna.

Det inversa fallet kan göras i två steg, först söka sig tillbaka från TCP till handleden och sedan genom kunskap om handledens position ta reda på de tre vridningsaxlarna ϕ , θ och α .

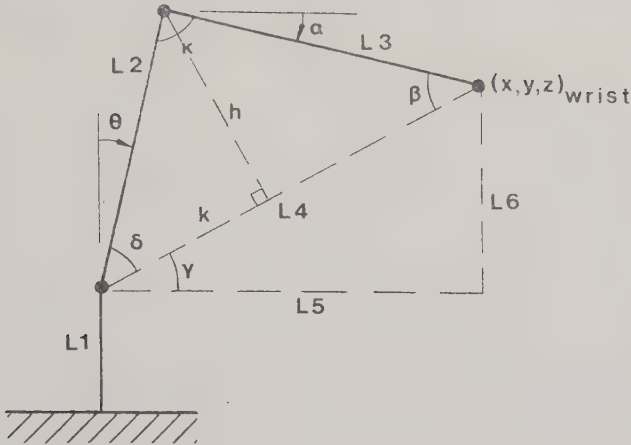
För det första steget är alla parametrar kända och man kan räkna sig tillbaka till handleden från arbetspunkten enligt följande: Beräkna avståndet från handledens koordinatsystem till TCP enligt (5-40) - (5-45). Överför sedan detta till baskoordinatsystemet genom att ta hänsyn till vridningen ϕ och man får

$$\begin{cases} X_{\text{wrist}} = X_{\text{TCP}} - p_x \cos \phi \\ Y_{\text{wrist}} = Y_{\text{TCP}} - p_y \sin \phi \\ Z_{\text{wrist}} = Z_{\text{TCP}} - p_z \end{cases} \quad (5-46)$$

där X_{wrist} , Y_{wrist} , Z_{wrist} anger handledens position i baskoordinatsystemet, X_{TCP} , Y_{TCP} , Z_{TCP} anger TCP i baskoordinatsystemet och p_x , p_y , p_z anger avståndet från handleden till TCP i handledens koordinatsystem.

För det andra steget är handledens position känd enligt (5-46) ovan och robotarmens ledpositioner för ϕ -, θ - och α -xlarna skall bestämmas.

För ϕ -axeln gäller att

Bild 5.10 Länksystem för θ - och α -axlarna.

$$\varphi = \arctan \frac{x_{\text{wrist}}}{y_{\text{wrist}}} \quad (5-47)$$

$$\text{där } \varphi \text{ definieras så att } \varphi = \begin{cases} 0 \leq \varphi \leq \frac{\pi}{2} & +X, +Y \\ \frac{\pi}{2} \leq \varphi \leq \pi & -X, +Y \\ -\pi \leq \varphi \leq -\frac{\pi}{2} & -X, -Y \\ -\frac{\pi}{2} \leq \varphi \leq 0 & +X, -Y \end{cases}$$

För θ - och α -axlarna erhålles ett länksystem enligt bild 5.10. Man ser ur bild 5.10 att

$$\theta = \frac{\pi}{2} - \delta - \gamma \quad (5-48)$$

och att

$$\alpha = \beta - \gamma \quad (5-49)$$

Vinkeln γ kan erhållas ur längderna $L5$ och $L6$ där

$$L5 = \sqrt{x_{\text{wrist}}^2 + y_{\text{wrist}}^2} \quad (5-50)$$

och

$$L6 = z_{\text{wrist}} - L1 \quad (5-51)$$

och man får att

$$\gamma = \arctan \frac{L6}{L5} \quad , \quad 0 \leq \gamma \leq \frac{\pi}{2} \quad (5-52)$$

Vinkeln δ kan uttryckas med hjälp av h och s .

Om vi sätter $s = (L2 + L3 + L4)/2$, erhålles arean A för triangeln $L2$ - $L3$ - $L4$ enligt Heron's formel som

$$A = \sqrt{s(s-L2)(s-L3)(s-L4)} \quad (5-53)$$

och höjden h i triangeln blir

$$h = \frac{2A}{L4} \quad (5-54)$$

där L_4 kan skrivas som

$$L_4 = \sqrt{X_{\text{wrist}}^2 + Y_{\text{wrist}}^2 + Z_{\text{wrist}}^2} \quad (5-55)$$

k kan beräknas som

$$k = \sqrt{L_2^2 - h^2} \quad (5-56)$$

och δ kan nu skrivas som

$$\delta = \arctan \frac{h}{k}, \quad 0 \leq \delta \leq \frac{\pi}{2} \quad (5-57)$$

Genom att utnyttja resultaten från (5-52) och (5-57) kan θ beräknas enligt (5-48). För vinkeln α återstår nu att beräkna β -vinkeln. Denna kan uttryckas som

$$\beta = \arctan \frac{h}{k-L_4}, \quad 0 \leq \beta \leq \frac{\pi}{2} \quad (5-58)$$

och α kan beräknas enligt (5-49) genom att utnyttja resultaten från (5-52) och (5-58).

För robotens vridningsaxlar gäller mekaniska begränsningar enligt följande (för armbågsvinkeln κ gäller enligt bild 5.10 att $\kappa = \pi/2 + \theta + \alpha$):

$$\left\{ \begin{array}{l} -170^{\circ} \leq \varphi \leq 170^{\circ} \\ -40^{\circ} \leq \theta \leq 40^{\circ} \\ -25^{\circ} \leq \alpha \leq 40^{\circ} \\ -90^{\circ} \leq \tau \leq 90^{\circ} \\ -180^{\circ} \leq \psi \leq 180^{\circ} \\ 50^{\circ} \leq \kappa \leq 130^{\circ} \end{array} \right.$$

vilka man måste ta hänsyn till vid praktiskt utnyttjande av transformationsfunktionerna.

5.3.3 NIVA 3 - GENERERING AV ENKLA ROBOTRÖRELSE

För att kunna definiera fysiska aktiviteter hos roboten krävs, att man har tillgång till en uppsättning av primitiva kommandon som kan generera enkla rörelser. Med enkla rörelser avses rörelser som kan definieras av en start- och slutposition och med definierad hastighet så att kommandot genereras som en kontinuerlig rörelse med avsedd funktion. Alla rörelser som tas upp nedan, förutsätter att rörelsekommandona kan anges med kartesiska koordinater som indata för definitionspositionerna. I praktiken innebär detta att man explicit anger TCP med X, Y, Z samt handledens orientering som vridningar av τ - och ψ -axlarna.

Om man börjar på nivå 1 i hierarkin, ser man att det endast finns robotinkrement som kan definiera en rörelse på denna nivå. En enkel rörelse i robotinkrement anges i fortsättningen som REV (REVolute movement) och innebär att rörelsen av robotaxlarna sker mot en slutposition på ett osynkroniserat sätt. Hastigheten anges för rörelseaxlarna i procent av maximal hastighet och varje axel kommer således att röra sig oberoende av de övriga mot sin slutposition. Karakteristiskt för rörelsen är en hög förflyttningshastighet mot priset av en svåröverskådlig och odefinierad rörelse i rummet. Indata till rörelsekommandot är aktuell position och önskad position i det kartesiska baskoodinatsystemet samt hastigheten för rörelseaxlarna i procent av maximal hastighet (± 127 robotinkrement/rörelse-

axel). Utifrån positionsdata beräknas den inkrementella förflyttning som rörelsen skall generera för respektive axel. Interpolationen mellan start- och slutpositionen kan sedan utföras på nivå 1 i rörelsehierarkin.

På nivå 2 i hierarkin finns möjlighet att definiera rörelsen utifrån robotens rörelseaxlar i robotkoordinater, dvs som vridningar av φ -, θ -, α -, τ - och ψ -axlarna. Rörelsen i robotkoordinater anges i fortsättningen som RES (REvolute Synchronized movement). RES-rörelsen innebär, att robotaxlarna rör sig från en start- till en slutposition på ett synkroniserat sätt så att samtliga axlar uppnår slutpositionen samtidigt. Karakteristisk för rörelsen är en hög förflyttningshastighet mot priset av en svåröverskådlig och odefinierad rörelse i rummet. Rörelsen har dock ett jämnare förlopp i rummet relativt REV-rörelsen ovan och ger över kortare sträckor en approximativ linjär rörelse i rummet. Indata till rörelsekommandot är aktuell position och önskad position i det kartesiska baskoordinatsystemet samt den tid som åtgår för att utföra rörelsen. Ett alternativ till tidsangivelsen är att ange hastigheten i mm/s för TCP. Den sträcka som TCP genomlöper under rörelsen kan sedan approximativt beräknas genom en linjärisering av rörelsen i ett antal segment som bestäms av den rörelseaxel som utför den största vridningen. Utifrån positionsdata beräknas de vridningar som rörelsen skall generera på respektive axel. Interpolationen mellan start- och slutpositionen kan sedan utföras på nivå 2 i rörelsehierarkin.

I nivå 3 i hierarkin definieras rörelserna utifrån det kartesiska baskoordinatsystemet. Den grundläggande rörelsen på denna nivå är en linjär rörelse i rummet, fortsättningsvis kallad LIN (LINEar movement). LIN-rörelsen innebär att TCP rör sig från en start- till en slutposition linjärt i rummet med definierad hastighet. Karakteristiskt för rörelsen är en begränsad hastighet och rörelsemöjlighet beroende på start- och slutposition men i gengäld erhålls en definierad rörelse i rummet. Indata till rörelsekommandot är aktuell position och önskad position i det kartesiska baskoordinatsystemet samt hastigheten för TCP i mm/s. Utifrån positions- och hastighetsdata interpoleras rörelsen linjärt över X-, Y- och Z-axlarna samt handledens vridningsvinklar τ och ϕ . För varje interpoleringspunkt längs rörelsebanan beräknas ny position i kartesiska koordinater, robotkoordinater och robotinkrement. Interpolationen mellan start- och slutpositionen utförs i nivå 3 i rörelsehierarkin. Beräkningsgången

av en linjär interpolering blir följande med given start- och slutposition, hastighet (v) samt interpolationsfrekvens (f):

1. Beräkna distansen mellan start- och slutpositionen och sätt den till $lindistance$.
2. Beräkna tiden för rörelsen som $lindistance/v$ och sätt den till $lintime$.
3. Beräkna den inkrementella förflyttningen av respektive koordinat för varje interpolationspunkt som $ds_i \leftarrow dl/(f*time)$, där dl är differensen mellan start- och slutpositionen för respektive koordinat under rörelsen, $0 \leq ds_i \leq 1$.
4. Beräkningen av en interpolationspunkt för respektive koordinat utföres som $l \leftarrow l + ds_i$, där l anger respektive koordinats aktuella position.

En annan rörelse som är vanligt förekommande i praktiska tillämpningar är cirkulär rörelse i rummet, fortsättningsvis kallad CIR (CIRcular movement). CIR-rörelsen innebär, att TCP rör sig från en start- till en slutposition cirkulärt i rummet med definierad hastighet. Indata till rörelsekommandot eller den information som krävs för att generera rörelsen är cirkelns centrum (X, Y, Z), radie (R), startvinkel och slutvinkel, lutning ξ kring X -axeln och η kring Y -axeln samt riktningsen av rörelsen längs cirkeln. Beräkningsgången av en cirkulär interpolering blir följande med indata enligt ovan:

1. Beräkna längden mellan start- och slutposition och sätt den till $cirdistance$.
2. Beräkna tiden för rörelsen som $cirdistance/v$ och sätt den till $cirtime$, v är hastigheten.

3. Beräkna den inkrementella vridningen längs cirkelrörelsen för varje interpoleringspunkt som $d\zeta \leftarrow dl/(f \cdot \text{cirtime})$, där dl är differensen mellan start- och slutvinkel för rörelsen längs cirkeln med rätt tecken relativt rörelseriktningen, $0 \leq |d\zeta| \leq 1$.

4. Interpolationspunkterna över cirkeln kan beräknas som

```

vinkel ← startvinkel
X ← R*cos(vinkel)
Y ← R*sin(vinkel)
vinkel ← vinkel + dz
(fortsätt tills vinkel=slutvinkel)

```

5. Transformera ut punkten på cirkeln med translationen X, Y, Z (cirkelcentrum) och rotationerna ξ kring X -axeln och η kring Y -axeln enligt tekniken med homogena transformationsmatriser ovan.

Beräkningen av CIR-rörelsen kräver enligt förfarandet ovan en stor mängd beräkningsoperationer under interpoleringen. Ett sätt att minska detta och samtidigt erhålla en generalitet vid interpolering av rörelser i kartesiska koordinater är att göra en uppdelning av den enkla rörelsen i segment som var för sig beskrivs som polynomfunktioner av tredje ordningen [5.3]. Tekniken kan kort beskrivas enligt följande:

Betrakta ett segment i av den rörelse som skall genomlöpas mellan tiden t_{i-1} och t_i . Positioneringen längs det i :te segmentet kan approximeras som

$$X(t) = A_i b(t) \quad (5-59)$$

där

$X(t)$ är en n -dimensionell vektor som beskriver rörelsen,

A_i är en $n \times 4$ koefficientmatris,

$$b(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}^t$$

t varierar från t_{i-1} till t_i .

på samma sätt kan positionsändringen längs det i :te segmentet approximeras med derivatan enligt

$$\dot{X}(t) = A_i \dot{b}(t) \quad (5-60)$$

$$\text{där } \dot{b}(t) = \begin{bmatrix} 3t^2 & 2t & 1 & 0 \end{bmatrix}^t$$

Koefficientmatrisen A_i kan sökas med olika kurvanslutningsmetoder. För robottillämpningar uppvisar kubiska splines lämpliga egenskaper i och med att man kan erhålla kontinuerliga övergångar mellan segmenten [5.4]. Om vi bortser från accelerationen i segmenten (accelerationen är oftast av litet intresse vid bågsvetsning eller vid de hastigheter som gäller för bågsvetsning), har vi 4 randvillkor som löser A_i -matrisen, position och hastighet vid varje ände av segmenten. Om positionen och hastigheten är given för tiderna t_{i-1} och t_i , kan A_i erhållas ur sambanden från (5-59) och (5-60) för varje segment i som

$$\begin{bmatrix} X(t_{i-1}) & \dot{X}(t_{i-1}) & X(t_i) & \dot{X}(t_i) \end{bmatrix} = A_i \begin{bmatrix} b(t_{i-1}) & \dot{b}(t_{i-1}) & b(t_i) & \dot{b}(t_i) \end{bmatrix} \quad (5-61)$$

och A_i erhålles som

$$A_i = \begin{bmatrix} X(t_{i-1}) & \dot{X}(t_{i-1}) & X(t_i) & \dot{X}(t_i) \\ b(t_{i-1}) & \dot{b}(t_{i-1}) & b(t_i) & \dot{b}(t_i) \end{bmatrix}^{-1} \quad (5-62)$$

Och man ser, att det för beräkning av k succesiva segment endast krävs t_i , $X(t_i)$ och $\dot{X}(t_i)$ för att bestämma rörelsen, $i=0, 1, 2, \dots, k$.

För att erhålla ett förenklat beräkningsförfarande införs en variabel s , $0 \leq s \leq 1$. Då man låter s gå från 0 till 1 då t varierar från t_{i-1} till t_i . t är då relaterad till s enligt

$$t = t_{i-1} + (t_i - t_{i-1}) \cdot s \quad (5-63)$$

och

$$\frac{dt}{ds} = t_i - t_{i-1} \quad (5-64)$$

Genom utnyttjande av detta förhållande kan man skriva (5-59) och (5-60) som

$$X(s) = A_i' b(s) \quad (5-65)$$

och

$$\dot{X}(s) = A'_1 \dot{b}(s) \quad (5-66)$$

$$\text{där } b(s) = \begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix}^t$$

$$\text{och } \dot{b}(s) = \begin{bmatrix} 3s^2 & 2s & 1 & 0 \end{bmatrix}^t$$

Randvillkoren vid ändpunkterna vid det i:te segmentet blir då

$$\left. \begin{array}{ll} \text{vid } t=t_{i-1}: s=0, & X(s) = X(t_{i-1}) \\ & \dot{X}(s) = \dot{X}(t_{i-1}) \frac{dt}{ds} \\ \text{vid } t=t_i : s=1, & X(s) = X(t_i) \\ & \dot{X}(s) = \dot{X}(t_i) \frac{dt}{ds} \end{array} \right\} \quad (5-67)$$

Koefficientmatrisen A'_1 kan beräknas i analogi med (5-62) ur (5-65) och (5-66) som

$$A'_1 = \begin{bmatrix} X(t_{i-1}) & \dot{X}(t_{i-1}) \frac{dt}{ds} & X(t_i) & \dot{X}(t_i) \frac{dt}{ds} \\ b(0) & \dot{b}(0) & b(1) & \dot{b}(1) \end{bmatrix}^{-1} \quad (5-68)$$

Matrisinversionen är här enkel att utföra och kan förberäknas som

$$\begin{bmatrix} b(0) & \dot{b}(0) & b(1) & \dot{b}(1) \end{bmatrix}^{-1} = \begin{bmatrix} 0 & 0 & 1 & 3 \\ 0 & 0 & 1 & 2 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}^{-1} =$$

$$\begin{bmatrix} 2 & -3 & 0 & 1 \\ 1 & -2 & 1 & 0 \\ -2 & 3 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \quad (5-69)$$

Positionen längs det i:te segmentet kan nu beräknas vid tiden t med s som

$$s = \frac{t - t_{i-1}}{t_i - t_{i-1}} \quad (5-70)$$

eller då man vid interpolering vill generera en serie punkter längs konturen med intervallet Δt som

$$\Delta s = \Delta t \frac{ds}{dt} = \frac{\Delta t}{t_i - t_{i-1}} \quad (5-71)$$

och man erhåller en serie interpolationspunkter längs det i:te segmentet genom att utnyttja (5-65) för s går från 0 till 1 med stegstorlek som bestäms av önskad hastighet.

Den beskrivna metoden med approximerade kurvsegment innebär en generaliserad teknik för interpolering av robotens rörelser.

För LIN-rörelser erhålls en exakt beskrivning av rörelsen i ett segment. En intressant variant av LIN-rörelsen är en VEC-rörelse (VECTor), där man endast specificerar rörelsens riktning och hastighet. Genom information från externa sensorer kan man på detta sätt låta yttre händelser ge indata till robot-systemet som under exekvering bestämmer dess utseende, t ex fögföljning vid svetsning.

Vid bågsvetsning är det önskvärt att kunna åstadkomma en pendlande rörelse under svetsning. Pendlingen utförs så att en rörelse vinkelrät mot huvudriktningen överlagras den egentliga rörelsen. Så länge pendlingen utförs på raka fogar är problemet trivialt genom att pendlingsrörelsen är konstant under svetsrörelsen. För det generella fallet krävs att pendlingen skall utföras i XY-planet_{TCP} (se bild 5.9) vinkelrät mot svetsrörelsen. För att erhålla en kontrollerad pendling vrides koordinatsystemet i TCP kring sin Z-axel så att X-axeln_{TCP} sammanfaller med den aktuella svetsriktningen. Y-axeln_{TCP} kommer då att vara vinkelrät mot svetsrörelsen och definiera pendlingsriktningen.

För CIR-rörelsen erhålls en approximativ lösning, där varje segment med en viss noggrannhet beskriver en del av cirkeln. Maximala felet på segmentet är proportionellt mot radien och segmentets storlek. Om ett fel om högst 0.1mm tillåtes, erhålles ett samband mellan radie och segmentlängd enligt följande:

segmentlängd (rad)	radie (mm)
$\pi/2$	6
$\pi/4$	100
$\pi/8$	1600
$\pi/16$	26000
$\pi/32$	410000

Man ser, att radien ökar ca 16 gånger för en halvering av segmentlängden och att det för normala arbetsområden endast kommer att krävas 8-16 segment för att fullborda en hel cirkel. För varje cirkelrörelse kan A-matriserna förberäknas och man får på så sätt ett mindre beräkningsarbete under interpoleringen av kurvsegmenten.

En ytterliggare konsekvens av den generella interpoleringstekniken enligt ovan är de utökade möjligheterna att direkt ta emot geometriinformation från CAD-definierad geometri. För element eller rörelser definierade som friformiga kurvor får man en exakt beskrivning då dessa sammanfaller med kubiska splines. I övriga fall kan normalt en mycket god approximation erhållas genom en överföring av rörelsen till kubiska splines under kontroll av noggrannheten.

En alternativ interpoleringsmetod som genererar definierade rörelser i rummet är att utnyttja tekniken ovan för att dela upp en rörelse i rummet till delsegment. För varje segment utför man sedan en kurvanpassning av tredje ordningen för rörelsen av varje axel i robotkoordinater [5.5]. Man kan på så sätt reducera beräkningsarbetet under interpoleringen genom att undvika att utföra transformationer mellan kartesiska koordinater och robotkoordinater. Beräkningen av segmenten och dess koeficientmatriser kan ske antingen on line eller off line. Vid on line beräkning kommer emellertid samplingsfrekvensen att reduceras drastiskt, om man skall få en mätbar minskning av beräkningsarbetet. Vid off line beräkning av segmenten och kurvanpassning av rörelsen i robotkoordinater erhåller man en rörelsegenerering som utför enkla rörelser i rummet på samma sätt som vid interpolering i kartesiska koordinater. Vid återkoppling av rörelsen från externa sensorer som t ex fogföljning vid svetsning har man vid denna interpoleringsteknik liten möjlighet att påverka rörelsen på önskat sätt. De krav som ställs är kunskap om hur en avvikelse från rörelsebanan mätt i sensors koordinatsystem skall översättas till en rörelse i robotkoordinater. Om man vid applikationer som bågsvetsning har en önskan att kunna återkoppla rörelsen genom sensorer, ställer detta krav på kunskap om relationerna mellan olika koordinatsystem i realtid för att kunna styra TCP mot önskat läge. I och med detta kommer man att förlora det inbesparade beräkningsarbetet som skulle vara motivet för att använda den beskrivna metoden. Tekniken att interpolera i robotkoordinater

kan alltså inte anses lämplig vid applikationer som kräver återkoppling av rörelsen i realtid.

Vid generering av svetsrörelser utefter fogar är det av stor vikt att rörelsen utförs med hänsyn till krav som ställs från processen. Exempel på processparametrar som bestämmer svetsrörelsen och dess toleransområde är

- fogorientering
- pistolvinkel relativt objektet
- pistolvinkel relativt hastighetsvektorn
- fogtyp
- spänning och ström

Genom de krav som processen ställer på svetsrörelsen och objektets orientering krävs för flexibel produktion vid svetsning ofta en lägesställare som kan manipulera objektet i samarbete med robotens rörelser. Avsikten med detta förfarande är att robotens hantering av svetspistolen skall generera en önskad svetsrörelse på ett rörligt objektet, där styrningen av robotens axlar och lägesställarens axlar sker integrerat och synkroniserat med varandra. Man ser då att om vi bibehåller TCP relaterad till ett baskoordinatsystem i roboten kommer en linjär svetsrörelse längs en svetsfog ej att sammanfalla med fogen, om man samtidigt med robotrörelsen vrider lägesställaren för att säkerställa krav från processen som t ex fogorientering. Motivet att röra objektet under svetsning kan också gälla orienteringssvårigheter (speciellt vid 5-axliga robotar) eller åtkomlighet på objektet.

En samtidig rörelse av robot och objekt kräver en interpoleringsteknik som är objektorienterad, dvs interpoleringen sker utifrån ett koordinatsystem som är fast relaterat till objektet. Detta innebär, att en linjär rörelse längs objektet i själva verket kommer att generera en kurvformig rörelse i rummet beroende på hur objektet rör sig i rummet. För att erhålla en sådan interpoleringsteknik krävs en överföringsfunktion som transformerar koordinater beskrivna i objektets koordinatsystem till baskoordinatsystemet. Vid interpolering längs en kurva i objektets koordinatsystem kommer varje punkt som man interpolerar fram att överföras till ett koordinatvärde i baskoordinatsystemet och därefter till robotkoordinater osv enligt tidigare. För linjära förflyttningar

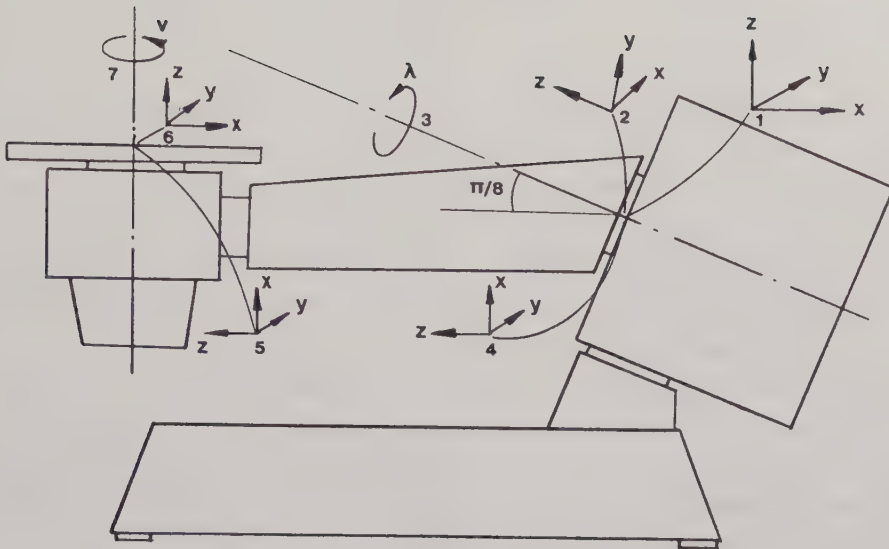


Bild 5.11 Principiell uppbyggnad av 2-axlig lägesställare typ ESAB ORBIT 500.

som överensstämmer med baskoordinatsystemets axelriktningar får man relativt enkla translationer att ta hänsyn till. För att kunna ta hänsyn till processen vid genereringen av rörelserna är det rimligt att anta att vi måste kunna vrida objektet kring flera axlar för att uppnå lämpliga orienteringar och positioner.

Som exempel på en svetsrobotstation med rörligt objekt kommer här en svetsrobotstation som förutom robot enligt tidigare dessutom består av en lägesställare av typ ESAB ORBIT 500 att behandlas. Lägesställarens uppbyggnad och rörelseaxlar framgår av bild 5.11. Lägesställaren har två rörelseaxlar av rotationstyp, λ och v . λ -axeln är vinklad $\pi/8$ i förhållande till horisontalplanet och med rotationsarmen vinklad $\pi/8$ i förhållande till λ -axeln så att rotationsarmen vid $\lambda=0$ ligger i horisontalplanet och vid $\lambda=\pi$ ligger $\pi/4$ mot horisontalplanet. v -axeln är vinklad $\pi/2$ mot λ -axelns rotationsarm. Detta

innebär, att man då $\lambda=0$ erhåller en fästplatta för fixtur och objekt som befinner sig i horisontalplanet och kan rotera kring sin z-axel som ligger i vertikalplanet.

Vid interpolering av en svetsrörelse antas att start- och slutposition är kända (objektkoordinater) genom t ex teach-in programmering och att vinklarna på lägesställaren och robotens handled därmed är kända. Antag vidare att man kan tillåta en linjär interpolering i vinkelkoordinater för lägesställaren som vid robotens handled enligt tidigare. Eftersom överföringsfunktionen mellan en position i baskoordinatsystemet och robotkoordinater är definierad enligt tidigare måste man nu söka uttrycka en position på lägesställaren $p(\lambda, \nu)$ i baskoordinatsystemet. Följande lösningsmetodik kan användas, där man med hjälp av homogena transformationsmatriser succesivt söker aktuell position. Koordinatsystemen numreras 0, 1, 2, ..., n räknat från baskoordinatsystemet. Skrivsättet $C\alpha$ resp $S\alpha$ anger cosinus respektive sinusfunktionerna för vinkeln α , där α anger godtycklig vinkel i radianer.

Börja med att lägga ut ett koordinatsystem i skärningen mellan λ -armen och dess rotationsaxel. Systemet transleras $(X, Y, Z)_1$ i förhållande till baskoordinatsystemet och vrides vinkeln ζ kring sin Z-axel. Vinkeln ζ kan beräknas som

$$\zeta = \arctan \left[\frac{Y_1}{X_1} \right] \quad (5-72)$$

, där $-\pi/2 < \zeta < \pi/2$ och $x > 0$

och man erhåller transformationsmatrisen som

$$T_{01}^1 = \begin{bmatrix} C\zeta & -S\zeta & 0 & X_1 \\ S\zeta & C\zeta & 0 & Y_1 \\ 0 & 0 & 1 & Z_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-73)$$

Koordinatsystemet vrids så att z_2 -axeln sammanfaller med rotationen λ . Detta ger en rotation kring y_1 -axeln med vinkeln $\mu=3\pi/8$ och man erhåller:

$$T_{11}^2 = \begin{bmatrix} C\mu & 0 & S\mu & 0 \\ 0 & 1 & 0 & 0 \\ -S\mu & 0 & C\mu & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-74)$$

Koordinatsystemet vrids därefter vinkeln λ kring z_2 -axeln och man erhåller:

$$T_{21}^3 = \begin{bmatrix} C\lambda & -S\lambda & 0 & 0 \\ S\lambda & C\lambda & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-75)$$

Koordinatsystemet vrids vinkeln $\iota = -\pi/8$ kring y_3 -axeln för att förbereda för translering längs λ -armen:

$$T_3^4 = \begin{bmatrix} C_\iota & 0 & S_\iota & 0 \\ 0 & 1 & 0 & 0 \\ -S_\iota & 0 & C_\iota & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-76)$$

Translater ut koordinatsystemet längs z_4 -axeln med längden z_5 (=längden på rotationsarmen λ) och längs x_4 -axeln med längden x_5 (upp till fästplattan för v -axeln):

$$T_4^5 = \begin{bmatrix} 1 & 0 & 0 & x_5 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & z_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-77)$$

Vrid upp koordinatsystemet så att z_6 -axeln blir rotationsaxel för v , dvs rotera systemet $\pi/2$ kring y_5 -axeln:

$$T_5^6 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-78)$$

Transformationen är nu framme vid fästplattan och man kan rotera koordinatsystemet kring dess z-axel med vridningen v som bestäms av vridningsaxelns position:

$$T_6^7 = \begin{bmatrix} C_v & -S_v & 0 & 0 \\ S_v & C_v & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-79)$$

För att överföra en godtycklig punkt $p(\lambda, v)$ på lägesställare till baskoordinater krävs slutligen en translering X_8, Y_8, Z_8 i förhållande till föregående koordinatsystem:

$$T_7^8 = \begin{bmatrix} 1 & 0 & 0 & X_8 \\ 0 & 1 & 0 & Y_8 \\ 0 & 0 & 1 & Z_8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-80)$$

Överföringsmatrisen mellan baskoordinatsystemet och v -axeln på lägesställaren erhålls genom att multiplicera ihop matriserna som

$$T_0^7 = T_0^1 \cdot T_1^2 \cdot T_2^3 \cdot T_3^4 \cdot T_4^5 \cdot T_5^6 \cdot T_6^7 \quad (5-81)$$

och man erhåller följande resultat för orienteringen och positioneringen av $(XYZ)_7$ -systemet relativt baskoordinatsystemet:

$$x_{7x} = C_v [- (S_i (C\zeta C_\mu C\lambda - S\zeta S\lambda) + C_i C\zeta S_\mu)] + S_v [- (C\zeta C_\mu S\lambda + S\zeta C\lambda)] \quad (5-82)$$

$$x_{7y} = C_v [- (S_i (S\zeta C_\mu C\lambda + C\zeta S\lambda) + C_i S\zeta S_\mu)] + S_v [- S\zeta C_\mu S\lambda + C\zeta C\lambda] \quad (5-83)$$

$$x_{7z} = C_v [S_\mu C\lambda S_i - C_\mu C_i] + S_v S_\mu S\lambda \quad (5-84)$$

$$y_{7x} = S_v [S_i (C\zeta C_\mu C\lambda - S\zeta S\lambda) + C_i C\zeta S_\mu] - C_v [C\zeta C_\mu S\lambda + S\zeta C\lambda] \quad (5-85)$$

$$y_{7y} = S_v [S_i (S\zeta C_\mu C\lambda + C\zeta S\lambda) + C_i S\zeta S_\mu] + C_v [- S\zeta C_\mu S\lambda + C\zeta C\lambda] \quad (5-86)$$

$$y_{7z} = S_v [- S_\mu C\lambda S_i + C_\mu C_i] + C_v S_\mu S\lambda \quad (5-87)$$

$$z_{7x} = C_i [C\zeta C_\mu C\lambda - S\zeta S\lambda] - S_i C\zeta S_\mu \quad (5-88)$$

$$z_{7y} = C_i [S\zeta C_\mu C\lambda + C\zeta S\lambda] - S\zeta S_\mu S_i \quad (5-89)$$

$$z_{7z} = - [S_\mu C\lambda C_i + C_\mu S_i] \quad (5-90)$$

$$X_7 = X_5 [C_1 (C\zeta C\mu C\lambda - S\zeta S\lambda) - S_1 C\zeta S\mu] + Z_5 [S_1 (C\zeta C\mu C\lambda - S\zeta S\lambda) + C_1 C\zeta S\mu] + X_1 \quad (5-91)$$

$$Y_7 = X_5 [C_1 (S\zeta C\mu C\lambda + C\zeta S\lambda) - S\zeta S\mu S_1] + Z_5 [S_1 (S\zeta C\mu C\lambda + C\zeta S\lambda) + C_1 S\zeta S\mu] + Y_1 \quad (5-92)$$

$$Z_7 = X_5 [-S\mu C\lambda C_1 - C\mu S_1] + Z_5 [-S\mu C\lambda S_1 + C\mu C_1] + Z_1 \quad (5-93)$$

Objektet som placeras på lägesställarens fästplatta kan antas vara definierat i $(XYZ)_7$ -systemet. Om en punkt P på objektet definieras med koordinaterna X_8 , Y_8 , Z_8 relaterat till $(XYZ)_7$ -systemet, får man P:s koordinater uttryckta i baskoordinatsystemet som:

$$P_x = X_8 \cdot X_{7x} + Y_8 \cdot X_{7y} + Z_8 \cdot X_{7z} + X_7 \quad (5-94)$$

$$P_y = X_8 \cdot Y_{7x} + Y_8 \cdot Y_{7y} + Z_8 \cdot Y_{7z} + Y_7 \quad (5-95)$$

$$P_z = X_8 \cdot Z_{7x} + Y_8 \cdot Z_{7y} + Z_8 \cdot Z_{7z} + Z_7 \quad (5-96)$$

Ekvationerna (5-94) - (5-96) kan på matrisform skrivas som

$$\begin{bmatrix} P_x - X_7 \\ P_y - Y_7 \\ P_z - Z_7 \end{bmatrix} = \begin{bmatrix} X_{7x} & X_{7y} & X_{7z} \\ Y_{7x} & Y_{7y} & Y_{7z} \\ Z_{7x} & Z_{7y} & Z_{7z} \end{bmatrix} \begin{bmatrix} X_8 \\ Y_8 \\ Z_8 \end{bmatrix} \quad (5-97)$$

Om matriserna i ekvation (5-97) betecknas så, att kolonnmatrisen i vänsterledet är ΔP_{bas} , 3x3-matrisen i högerledet A och kolonnmatrisen i högerledet P_{objekt} , erhålls objektkoordinaterna från baskoordinatsystemet som

$$P_{objekt} = A^{-1} \cdot \Delta P_{bas} \quad (5-98)$$

Genom att utnyttja (5-98) kan objektets koordinater erhållas vid programmering på objekt vid roboten. Överföringsfunktionerna beskrivna av (5-94) - (5-96) ger sedan möjlighet att interpolera rörelserna i ett koordinatsystem som är knutet till objektet. Man kan för varje interpolerad punkt på objektet överföra denna till koordinater i bassystemet. De erhållna överföringsfunktionerna innebär, att man kan utföra rörelser med en arbetspunkt relativt objektet istället för att utföra rörelsen relativt roboten själv som är det normala på dagens robotsystem utom då man tar hänsyn till enkla translationer, t ex löpande band etc. I och med detta övergår styrsystemet från att vara robotorienterat till att vara objektorienterat och vi skall härnäst se vad detta innebär vid generering av högre funktioner i systemet.

5.3.4 NIVA 4 OCH 5 - GENERERING AV HÖGRE FUNKTIONER

De högre funktionerna i rörelsehierarkin berör planering, optimering och generering av svetsrörelser på objektet, dvs automatisk programmering av svetsrörelsen med hänsyn tagen till processen. Medan beräkningarna i nivå 1, 2 och 3 till största delen är av robotnära karaktär med realtidskrav på i storleksordningen 10 ms, har vi här en planerande funktion som i de flesta fall kan utföras vid annan tidpunkt än själva svetsoperationen. Ett undantag för detta är då man vill kunna sensoråterkoppla även de högre funktionerna i systemet och genom detta automatiskt kunna ändra svetsförloppet på grundval av externa händelser. Valet av hur svetsrörelserna skall beräknas, off line eller on line, bestäms i stor utsträckning av användarfilosofi, systemuppbyggnad och användningssätt. För generering av svetsrörelsen måste man emellertid konstatera att det idag inte är praktiskt genomförbart att beräkna

denna on line till en rimlig datorkostnad.

Vid off line beräkning av svetsrörelserna har man möjlighet att manuellt styra eller ändra beräkningsgången och resultatet av beräkningarna. Vid on line beräkning av rörelserna däremot faller denna möjlighet bort och man måste kräva av systemet att det skall kunna göra alla beräkningar på ett korrekt sätt eller då lösningen ger ett dåligt resultat eller inget alls kunna tala om att så är fallet. Omvänt kan man också säga, att off line programmering av robotar för bågsvetsning kräver datorstöd vid beräkning och optimering av svetsrörelserna på grund av den stora komplexitetsgraden som uppstår i system med flera manipulatorer kopplade till varandra. Vid on line programmering finns inget sådant krav, utan man kan utnyttja systemet på traditionellt sätt, men erhåller en minskad programmeringstid genom det objekt-orienterade interpoleringsförfarandet beskrivet i avsnittet ovan för en viss typ av lägesställare.

För att motivera en lösning av det beskrivna problemkomplexet måste vi först ställa frågan:

Varför används manipulatorer i svetsrobotsystem och måste rörelserna mellan roboten och manipulatern integreras till en enhet?

Användningen av manipulatorer vid robotsvetsning har två orsaker, dels att utöka robotens arbetsområde till områden som ligger utanför robotens räckvidd, dels att skapa de relationer mellan objekt och svetspistol som ger lämpliga pistolvinklar relativt grundmaterialet och hastighetsvektorn så att processen utförs på ett riktigt sätt. Vid orienteringen av objektet kan det dessutom ställas krav på svetsfogens orientering relativt horisontalplanet. Detta gör att man inte kan lösa orienteringsproblemet med fler rörelseaxlar på roboten, utan man blir tvungen att utnyttja ett system som innehåller flera manipulatorer. Om man vid svetsning av en fog har krav på olika orienteringar av objektet eller på grund av robotens begränsade räckvidd måste förflytta objektet under svetsning och samtidigt inte vill avbryta svetsoperationen för omstart, uppstår kravet på en integrerad rörelse av robot och manipulator.

Anledningen till att man vill undvika omstart är risk för tändfel, kallstart, läckage samt tidsförlust. Ett sätt att lösa detta är att som i dagens system utföra rörelsen utan kontroll på hur rörelsen sker relativt objektet. Detta förfaringssätt leder till en ökad programmeringstid genom ett ökat antal programmerade punkter på rörelsen för att erhålla en approximativt korrekt svetsrörelse samtidigt som man inte har någon kontroll över hur svetspistolen är orienterad mot objektet. För att erhålla ett fullständigt integrerat system bör man i stället låta interpoleringen ske på objektet, dvs på samma sätt som om objektet var fixerat i tid och rum till baskoordinatsystemet. Genom denna interpolationsteknik erhåller man vid denna typ av system en kraftigt reducerad programmeringstid jämfört med konventionell teknik.

Om vi anser oss motiverade att söka en lösning på problemet automatisk beräkning av svetsrörelser för en robotsvetsstation, uppstår frågeställningen

Vilka kriterier skall gälla för att vi skall anse att en framtagna svetsrörelse är korrekt med hänsyn till processen, dvs kan man ange en optimal rörelse samt hur rörelsetoleranserna för denna varierar med ändrade processdata?

För att besvara denna frågeställning krävs kunskap om vilken information som krävs för att generera en svetsrörelse i ett beskrivet system och därefter en beskrivning av hur denna information skall relateras till den aktuella processen. För att kunna generera en svetsrörelse krävs kunskap om manipulatorernas start- och slutposition, typ av rörelse (t ex linjär), orientering av svetspistolen relativt objektet under rörelsen samt förflyttningshastighet relativt objektet.

Vid bestämning av start- och slutpositionen måste man ställa kraven från processen (t ex horisontellt läge) i relation till de möjligheter som manipulatorsystemet har för att genomföra svetsrörelsen på ett för processen riktigt sätt (arbetsområde, orientering). Av detta följer att man vid generella objekt och i ett specifikt svetsrobotsystem inte kan räkna med att uppnå optimala rörelser vid svetsoperationerna på grund av mekaniska begränsningar. Det

intressanta vid automatisk generering av svetsrörelser är därför att utifrån en optimal rörelse bestämma de toleransområden för rörelsen, där man kan anse sig erhålla ett önskat resultat.

När det gäller tolerans i läge brukar man ofta som en tumregel säga att bågens aktiva del ligger över ett område som är lika med tråddiametern. I och med detta kan man anse att toleransen i position skall ligga inom $\pm 0.5 \times \text{tråddiametern}$. Noggrannhetskravet måste dessutom ställas i relation till fogens utseende och smältans storlek, där grövre fogar som regel kräver mindre noggrannhet och tvärtom. De toleranskrav vi ställer på rörelsen kommer också att bestämmas av toleranserna på objektet och dess enskilda detaljer. För en given största tillåten tolerans i rörelsen mellan svetspistol och objekt gäller att en noggrann svetsrörelse medger en grövre tolerans på objektet och tvärtom (förutsatt lika krav på slutprodukten). För resultatet gäller dock, så länge man ligger inom lägestoleranserna, att fogförberedningen måste ha utförts på ett korrekt sätt och att fixturerna har konstruerats på ett sådant sätt att man bibehåller önskade toleranser under svetsoperationerna på objektet.

För att erhålla en kontrollerad tolerans i rörelseoperationen måste vi också se till att processen kan utföras med en stabil ljusbåge som ger en kontrollerad och sprutfri övergång av material från tråden till grundmaterialet. Ett sätt att erhålla detta är att utnyttja transistoriserade strömkällor som arbetar enligt synergiprincipen med möjlighet att genom en pulsad ström generera en syntetisk spraybåge. Generellt kan man säga att dessa strömkällor ger ett säkrare val av svetsdata än traditionella strömkällor. Genom svetsförsök har man kunnat ta upp svetsdata som under bestämda förutsättningar ger säkra arbetsområden för hur svetsparametrarna skall väljas, se bild 5.12.

Om man för en situation enligt bild 5.12 väljer svetsdata på ett optimalt sätt relaterat till processen (t ex mitt i området, 1 droppe/puls), måste man ställa frågan, om detta automatiskt leder till ett minimum av krav på hur svetsrörelsen skall utföras för att erhålla ett önskat resultat. I verkligheten har man dessutom ingen kontroll över hur bra de valda svetsparametrarna är (open loop), utan man kan i ett oturligt fall ligga i utkanten av stabila svetsparametrar utan att upptäcka detta i processen. Om vi ställer bestämda

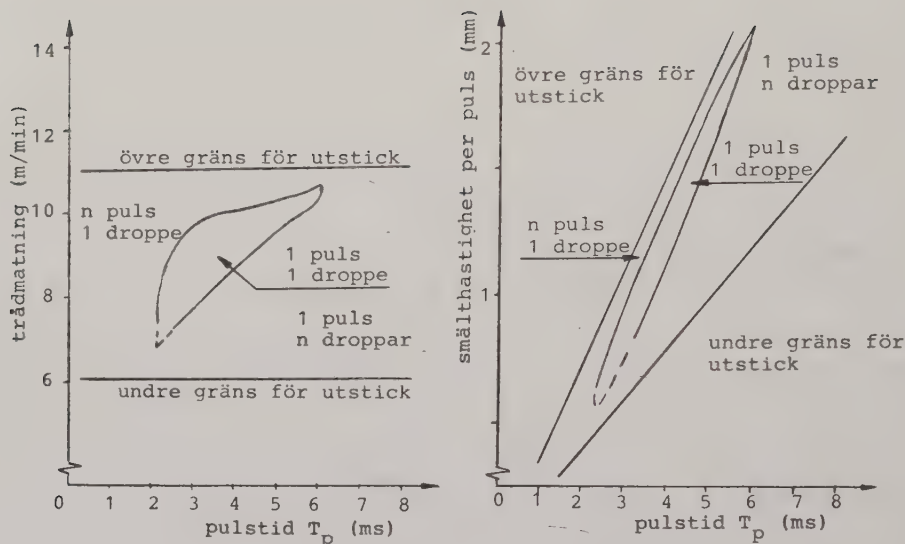


Bild 5.12 Exempel på stabila svetsdatazoner vid pulsad MIG/MAG-svetsning, 80%Ar+20%CO₂, 1.2mm tråd, $T_p = T_b$, $I_p = 350A$, $I_b = 50A$ [5.6].

krav på resultatet, kommer kraven på toleranserna i rörelsen att variera med hur svetsparametrarna väljs och hur rörelsen utförs. Beroende på rörelsen och objektets utseende kommer dessutom utsticket (avståndet från kontaktmunstycket till ljusbågen) att variera över svetsrörelsen. Om man inte återkopplar denna variation (svetspistolens höjd över fogen), kommer man att få en förändring av ljusbågen och materialtransporten som bl a beror på ökad förvärmning av tråden, skyddsgasens ändrade verkan, ökat spenningsfall i tråden och ändrade toleranser i trådens position.

Genom det förda resonemanget ovan kommer vi till slutsatsen, att valet av strömkälla, svetsparametrar, svetsrörelser på ett givet objekt och med önskat resultat skall ses som en helhet genom parametrarnas inbördes påverkan. Den för automatisk programmering av svetsrörelser intressanta frågeställningen är därför

Kan man ur givna förutsättningar (fogtyp, material, etc) bestämma optimala svetsdata och vilka toleranser som gäller för svetsrörelsen?

Optimala svetsdata avser här ett svetsdataval som inte bara är optimalt relativt processen utan även så att rörelsens toleransområde kan maximeras med bibehållet svetsresultat. Med utgångspunkt från att detta problem går att lösa kommer nedan en metodik att anges som för en svetsfog beräknar och genererar en önskad svetsrörelse inom givna toleransband. I kapitel sex skall en erfarenhetsbild ges av hur svetsparametrarna är relaterade till varandra och inom vilka arbetsområden som det beskrivna robotsystemet bör arbeta.

För orienteringen av svetsfogarna väljer vi en optimal orientering vilket i de flesta fallen kommer att sammanfalla med XY-planet_{bas}, dvs en vinkelrät vektor mot fogen som processvektor=(0, 0, -1). Svetsfogarnas orienteringar på objektet är beskrivna med motsvarande vektorer som pekar vinkelrät mot respektive svetsfog i objektets eller v-systemets koordinatsystem. Genom att utnyttja transformationsformlerna enligt (5-82) - (5-90) kan man för givna λ och v erhålla en orientering av v-systemet relativt baskoordinatsystemet. Orienteringen av fogvektorn i baskoordinatsystemet erhålls sedan utifrån v-systemets orientering som:

$$fog_{bas, x} = fog_{v, x} * v_{x, x} + fog_{v, y} * v_{y, x} + fog_{v, z} * v_{z, x} \quad (5-99)$$

$$fog_{bas, y} = fog_{v, x} * v_{x, y} + fog_{v, y} * v_{y, y} + fog_{v, z} * v_{z, y} \quad (5-100)$$

$$fog_{bas, z} = fog_{v, x} * v_{x, z} + fog_{v, y} * v_{y, z} + fog_{v, z} * v_{z, z} \quad (5-101)$$

där $fog_{bas, x}$ är fogvektorns x-komponent i bassystemet,
 $fog_{v, x}$ är fogvektorns x-komponent i v-systemet,
 $v_{y, x}$ är y-axelns_{v-syst} x-komponent i bassystemet.

För den framtagna fogvektorn_{bas} kan vinkeln φ mellan denna och processvektorn erhållas som:

$$\varphi = \arctan \left\{ \frac{|\text{fogvektor}_{\text{bas}} \times \text{processvektor}|}{\text{fogvektor}_{\text{bas}} \cdot \text{processvektor}} \right\} \quad (5-102)$$

där $-\pi/2 < \varphi < \pi/2$

Genom att kontrollera vektorernas riktning kan man beräkna den verkliga vinkeln mellan de båda vektorerna. Om vinkeln φ ligger inom ett tillåtet toleransområde, kan man anse att man har en positionering av objektet som utifrån processkraven ger en lämplig orientering.

För en bestämd fogvektor i v-systemets koordinatsystem och önskad processvektor i baskoordinatsystemet erhålles områden för lägesställarens rörelseaxlar som ger en differensvinkel mellan den önskade vektorn (processvektorn) och den erhållna (fogvektor_{bas}) inom specifika toleransband, se bild 5.13. Utifrån dessa områden skall robotens svetsrörelser beräknas. Om hela svetsrörelsen för en position av lägesställaren ligger inom robotens arbetsområde, är problemet löst genom att välja den inom robotens arbetsområde mest optimala position för lägesställaren. Svetsningen utförs sedan med roboten mot en stillastående lägesställare. Om man emellertid inte kan uppnå en godtagbar orientering av fogen med stillastående lägesställare så att svetsrörelsen ligger inom robotens arbetsområde, får man söka en samordnad rörelse av lägesställare och robot så att en svetsrörelse blir möjlig att utföra med bibehållna processkrav. Utifrån tillåten tolerans i orienteringen av objektet kommer man för en svetsrörelse att erhålla arbetsområden som roboten kan arbeta inom.

Då orienteringen av fogvektorn är bestämd, skall robotens orientering för handledsaxlarna τ och ψ sökas. Svetspistolens orientering skall med hjälp av handledsaxlarna med så stor noggrannhet som möjligt överrensstämmas med fogvektor_{bas}. Genom att utnyttja (5-41) och (5-43) erhåller man en transformationsmatris T_{TCP} för en svetspistol enligt

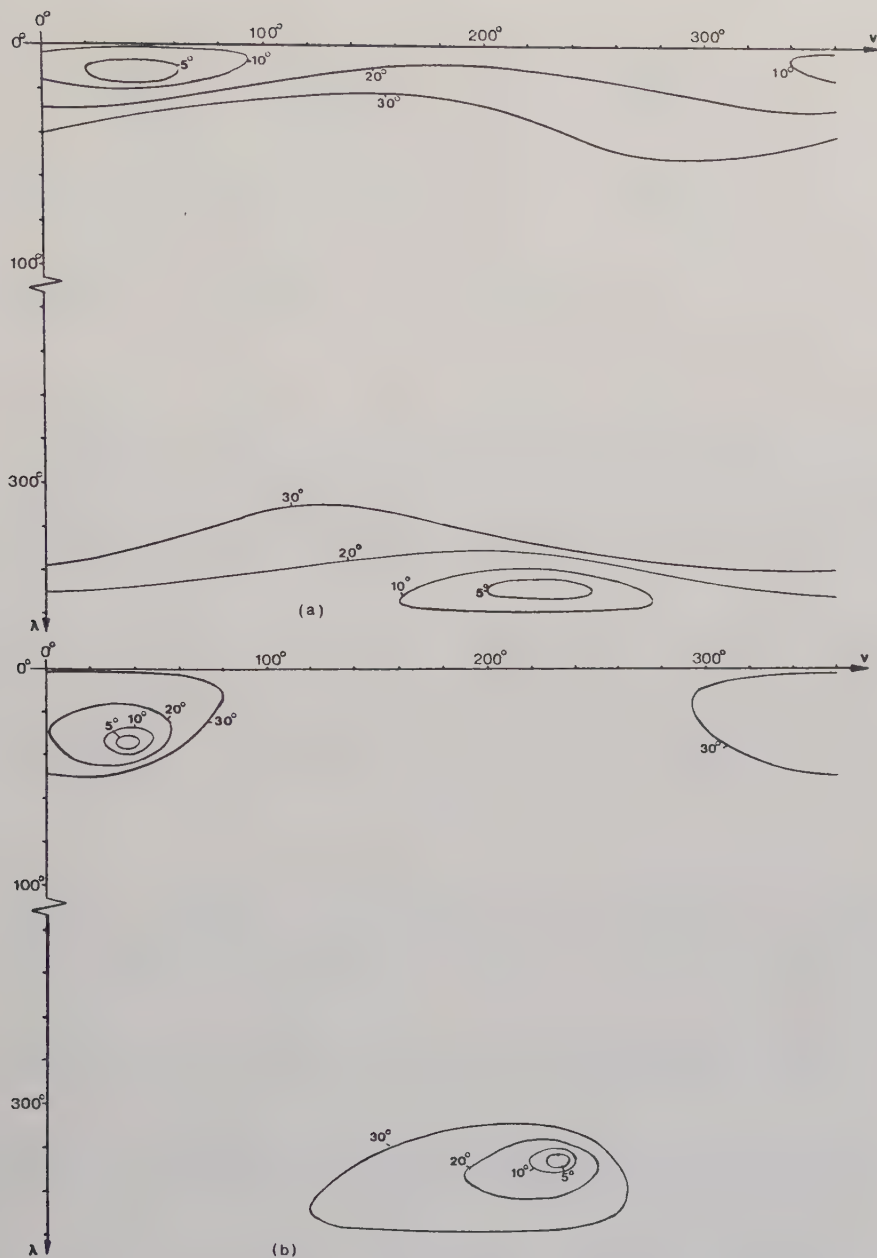


Bild 5.13 Differensvinkel (ϕ) mellan processvektor och fogvektor i baskoordinatsystemet för olika positioner av lägesställarens rörelseaxlar. Processvektor $= (0, 0, -1)$. (a) fogvektor $_v = (1, 1, -7)$, (b) fogvektor $_v = (3, 3, -7)$.

$$T_{\text{wrist}}^{\text{TCP}} = \begin{bmatrix} -S\tau C\psi C\eta & S\tau S\psi & -S\tau C\psi S\eta & -S\tau C\psi dx \\ -C\tau S\eta & & +C\tau C\eta & +(dz+L4)C\tau \\ S\psi C\eta & C\psi & S\psi S\eta & S\psi dx \\ -C\tau S\eta & C\tau S\psi & -C\tau S\eta & -C\tau dx \\ +S\tau S\eta & & -S\tau C\eta & -(dz+L4)S\tau \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(5-103)

där S och C betecknar sinus respektive cosinus, L4 är längden av handledens böjaxel, dz är avståndet från handledens fästplatta i ψ -axelns riktning till TCP, dx är avståndet från handledens fästplatta till TCP i negativ Z_{bas} -riktning och η är lutningen av svetspistolens räknat från handledens böjaxel.

Med insatta värden på $dx=60$, $dz=300$, $\eta=\pi/4$ och $L4=95$ kan man för en fogvektor $_{\text{bas}}$ som ligger inom 30° från vertikalvektorn $(0, 0, -1)$ alltid erhålla värden på τ och ψ så att svetspistolens orientering med säkerhet ligger inom storleksordningen en grad till fogvektorn $_{\text{bas}}$. I praktiken kommer sökningen av den samordnade rörelsen att bli en iterativ process, i de fall man erhåller rörelser som ligger utanför robotens arbetsområde.

Beroende på den frihetsgrad som finns i utformningen av objekt kan en mängd fall uppträda som försvårar en automatisk sökning av svetsrörelser. Bland dessa kan nämnas

- svetsfogarnas orientering och möjlighet att orientera dessa på lämpligt sätt
- objektets utformning vid svetsfogarna och den åtkomlighet som erhålls vid valda orienteringar
- svetsfogarnas utformning och möjlighet att åstadkomma en samordnad svetsrörelse

Svetsfogarnas orientering och möjlighet att orientera dessa med lägesställaren på önskat sätt bestäms av hur svetsfogarna har utformats på objektet, lägesställarens orienteringsmöjlighet samt hur objektet placeras på lägesställaren.

Objektets utformning bestämmer åtkomligheten vid svetsrörelserna och hur till- och frånrörelserna kan utföras. Om åtkomligheten är god i fogvektorns riktning mot fogen, kan till- och frånrörelserna bestämmas på motsvarande sätt som vid framtagning av verktygsrörelser vid fräsning, där man genom att ange clear plane, approach movement och retract movement automatiskt genererar dessa rörelser. Vid risk för kollision krävs en kontroll av hur rörelsen utförs och en möjlighet att manuellt definiera dessa rörelser.

Svetsfogarnas utformning på objektet bestämmer hur svetsrörelsen skall utföras. Vid korta, linjära fogar med konstanta fogvektorer kan lägesställaren med stor sannolikhet orientera objektet till en konstant position under en svetsrörelse. Om fogen har lång utsträckning, kommer roboten i många fall utanför sitt arbetsområde. Vid konstant fogvektor måste man därför söka en samordnad rörelse för lägesställaren inom de områden som beskrivs ovan och i bild 5.13 så att hela svetsrörelsen befinner sig inom robotens arbetsområde. Vid varierande fogvektor längs en fog måste rörelsen delas upp så att man för varje delsegment söker möjliga rörelseområden för lägesställaren och robotens samordnade rörelse. Man kan konstatera att kravet på kontroll av rörelsen (t ex visuellt via grafik) blir större för mera avancerade rörelser och objekt.

Enligt diskussionen ovan ser man att det för många objekt och svetsfogar är möjligt att automatiskt välja och generera en svetsrörelse för en beskriven svetsrobotstation. Vid en del typer av objekt eller svetsfogar kommer emellertid rörelsen att resultera i ett icke önskat förlopp genom t ex kollisioner vid till- och frånrörelser. Detta gör att valda svetsrörelser bör kontrolleras av en beredare. Detta förhållande medför att de högre funktionerna i styrsystemet inte kan betraktas som en del i realtids styrsystem sett i det tidsnära perspektivet utan att rörelserna för ett objekt planeras i sin helhet före svetsning utanför robotstationen.

För att en automatisk rörelsegenerering skall kunna ske måste tillräcklig information ges bland annat beträffande till- och frånrörelser, säkerhetsområden etc. Med tillgång till sådan information kan man förutom att generera svetsrörelser även analysera svetsrobotstationens möjlighet att svetsa ett definierat objekt.

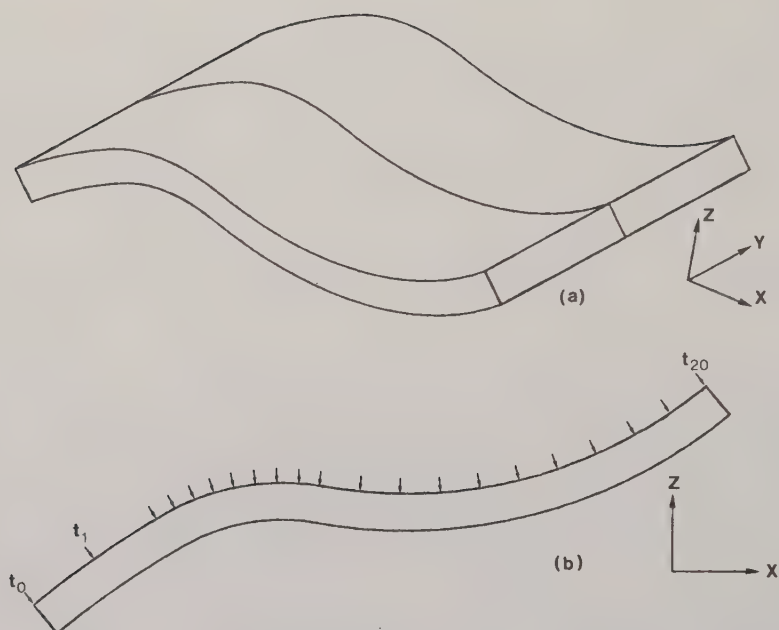


Bild 5.14 Objekt som skall svetsas i tvåmanipulatorsystem (a) samt uppdelning av fogen i segment med avseende på dess krökning (b).

Följande exempel illustrerar tillämpningen och resultatet av objekt- och processprogrammering:

Ett objekt skall svetsas enligt bild 5.14a. Objektets koordinatsystem är anpassat att sammanfalla med v -axeln enligt transformationsfunktionerna för ORBIT 500 enligt tidigare. Den valda processvektorn $(=0, 0, -1)$ skall uppnås med en tolerans av $\pm 5^\circ$.

Eftersom fogen har en krökning, krävs en uppdelning av denna i delsegment. För att klara toleranskravet avdelas delsegmenten, då fogens krökning inom ett delsegment överstiger 5° . Man erhåller en serie interpolationspunkter på objektet som definierar fogens orientering under svetsning, se bild 5.14b

För varje interpolationspunkt skall en lämplig orientering av lägesställaren bestämmas. Detta ger en lösning enligt bild 5.15.

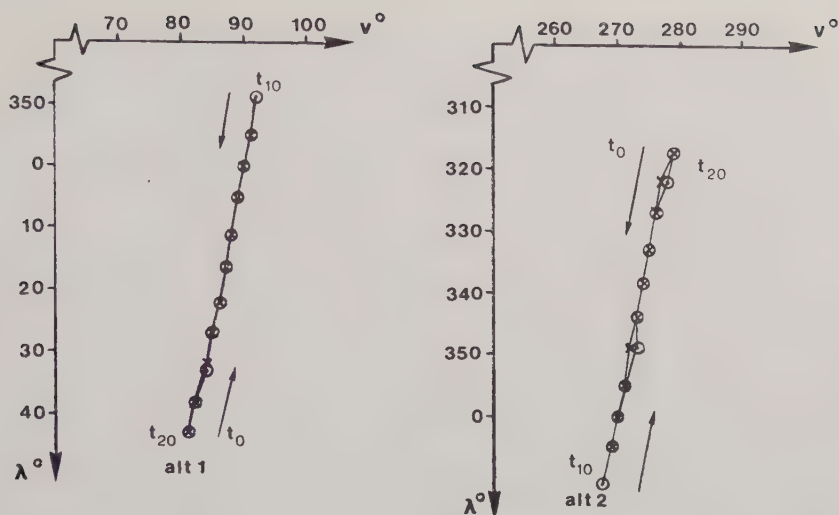


Bild 5.15 Lägesställarens rörelse under svetsning av objekt enligt bild 5.14.

I varje orienteringsläge och position av objektet skall en lämplig orientering av svetspistolen bestämmas. Genom detta har svetsrörelsen på objektet delats upp i segment. Rörelsen i varje segment utförs genom interpolering på objektet. I detta fall kan denna utföras med cirkulär interpolering. Den rörelse som roboten utför ges av bild 5.16. I det beskrivna objektet kan även till- och frånrörelserna definieras automatiskt genom angivelse av friplan kring objektet och till- och frånrörelsens riktning.

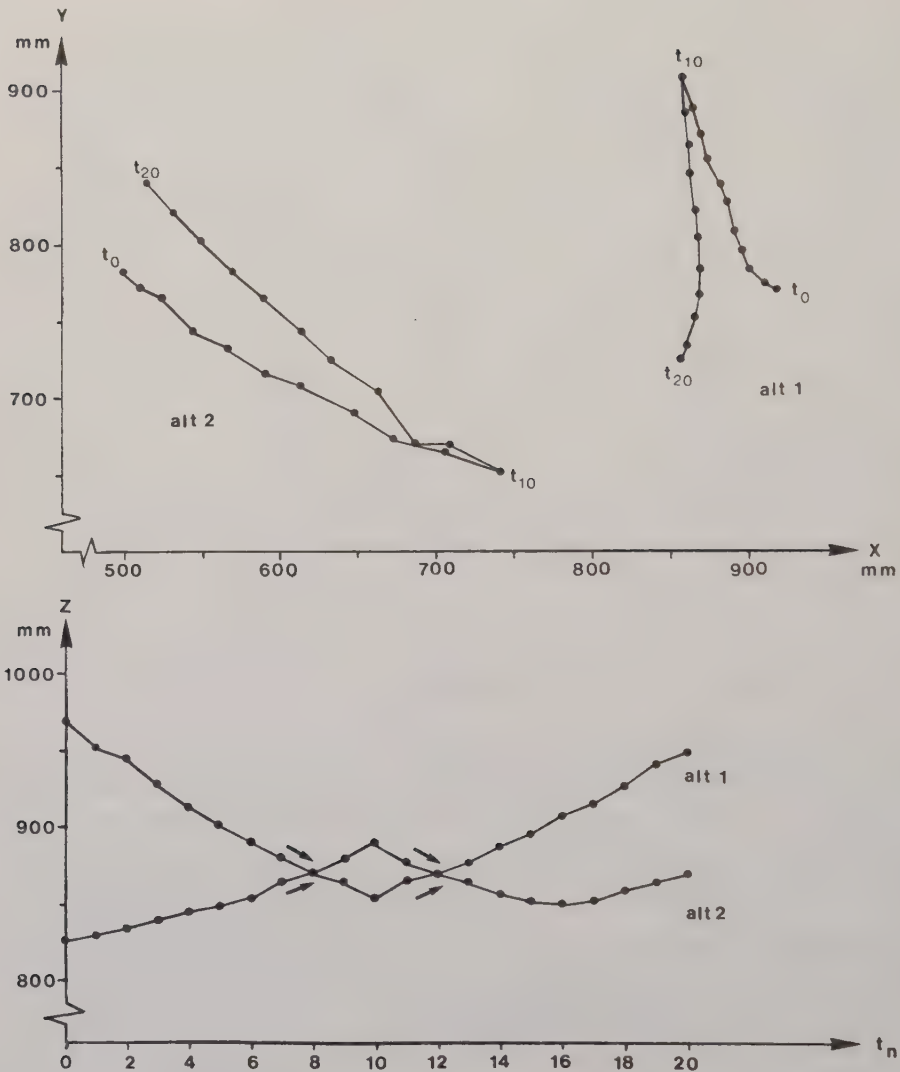


Bild 5.16 Robotens rörelse under svetsning av objekt enligt bild 5.14.

5.4 ATERKOPPLING AV SENSORINFORMATION

I robotstationen används sensorer för att mäta tillstånd på arbetsförlopp så att processen skall kunna utföras på ett korrekt sätt. För detta ändamål används interna och externa sensorer. De interna sensorerna används dels för att skydda roboten själv (t ex temperatur- och strömövervakning), dels för att

mäta robotaxlarnas positioner och hastigheter för att möjliggöra rörelser av det mekaniska systemet enligt definierade instruktioner. De externa sensorerna används för att anpassa och synkronisera robotens rörelser med omgivningen. För bågsvetsning är det förutom styrning och kontroll av strömkällan två tillstånd som är intressanta:

- objektets eller fixturens placering i robotstationen
- svetsfogarnas position och geometri

Objektets eller fixturens placering bestämmer, hur rörelseprogrammet skall orienteras eller transleras. Vid automatisk uppmätning av fixturens eller objektets position vid ofta återkommande tillverkning slipper man genom förflyttning av programmerade instruktioner utföra en tidsödande korrigering av svetsrörelserna.

Svetsfogarnas positioner och geometri ger möjlighet att före eller under svetsning korrigera rörelsen med hänsyn till programmerad rörelse.

I den följande texten behandlas inte själva sensortekniken för bågsvetsning (se avsnitt 4.2.4) utan hur sensorinformationen skall behandlas i systemet för att detta skall kunna anpassa robotens arbetssätt till en föränderlig omgivning. Vid uppmätning av geometriska avvikelser skall dessa överföras till baskoordinatsystemet. Detta innebär att om mätsensorn är monterad på svetspistolen, så erhåller man en lösning enligt det direkta kinematiska problemet i avsnitt 5.3.2.1. Transformationsmatrisen T_{TCP} kommer dock att få ett annat utseende eller numeriska värden beroende på hur sensorn är monterad på svetspistolen. Vid sensorteknik som utnyttjar ljusbågen, erhålls en lösning som är identisk med det direkta kinematiska problemet.

Den information som återkopplas skall därför anpassas till en nivå i rörelsehierarkin som arbetar med motsvarande information. Från robotsystemet och processen går därför en motsvarande hierarki med sensorinformation. På samma sätt som vid rörelsegenereringen är de lägre nivåerna robotorienterade, medan de högre nivåerna är objekt- eller processorienterade.

Från robotsystemet och processen samlas informationen in till nivå 1 i hierarkin. Vid användning av lägesgivare som skall ge information om geometriska avvikelser i processen kommer informationen att passera upp till nivå tre eller högre, där man arbetar i baskoordinater eller på objektet och processnivå. De lägre nivåerna kommer därför att behandla information om hur robotsystemetets enskilda delar utför de avsedda rörelserna, medan de högre nivåerna behandlar objekt- och processororienterad information. Det som avses i texten med sensorinformation är sådan information som är avgörande för att robotsystemet genom sina rörelser skall kunna fullgöra sin uppgift. Styrningen av strömkällan behandlas inte utom den ömsesidiga påverkan som svetsrörelsen kan ha på processen och tvärtom.

I nivå 1 i hierarkin samlas data in från olika sensorer som digitala värden eller digitaliseras via A/D-omvandlare. Behandlingen av informationen begränsas till en kalibrering av datavärdena till önskade enheter (läge i mm, vinkel i rad, etc). Den robotnära sensoråterkopplingen på nivå 1 behandlar de enskilda rörelseaxlarnas verkliga rörelser i inkrement liksom primära skyddssystem som t ex mätning av temperatur i styrskåpet och strömmar till drivdonen.

Nivå 2 arbetar med de enskilda ledernas rörelser i robotkoordinater. För att erhålla verklig position i robotkoordinater transformeras rörelseaxlarnas lägen uttryckta i inkrement till ledvinklar eller ledpositioner.

I nivå 3 transformeras robotsystemets ledvinklar och positioner till en position och orientering av TCP i baskoordinater enligt lösningen för det direkta kinematiska problemet. På samma sätt beräknas lägen och orienteringar i baskoordinater från sensorer som mäter hur svetspistolen ligger i förhållande till svetsfogen. Sensorinformation som inte enbart består av lägesangivelser utan bildar ett mönster eller beskrivning av hur processen utförs förs upp till nivå 4, där rörelsegenereringen är objekt- och processororienterad.

I de högre nivåerna, nivå 4 och 5, behandlas sådan information som kräver en väsentligt förändrad enkel robotrörelse för att processen skall utföras på ett korrekt sätt. Exempel på detta är om fogens geometri förändras så att det krävs en pendling i stället för rak rörelse, att fogen kräver ett annat antal svetssträngar än planerat för att erhålla rätt fyllnadsgrad eller att man vid uppmätning av objektets läge finner att det avviker så mycket att det krävs en omplanering av svetsrörelserna. Ett slutmål för en robotsvetsstation bör vara att kunna mäta och återkoppla information beträffande den valda svetsföljden, rörelsen och svetsdata för att ge önskat resultat i processen.

För att hålla isär informationsflödet skall sensorinformationen funktionsindelas enligt den princip som angetts i avsnitt 4.2.5. Beroende på de krav som en applikation ställer, kommer ett urval av dessa funktioner att utnyttjas. Vid bågsvetsning innebär t ex 'bestämma läge för detalj' att mäta upp position och eventuellt orientering av objekt eller svetsfog före svetsning, medan 'kontrollera manipuleringen' innebär att svetsrörelsen anpassas till fogens läge under svetsning. Man ser att informationen kräver olika aktiviteter i rörelsegenereringen. Det förstnämnda fallet innebär en omplanering av svetsrörelsen, medan det senare fallet succesivt anpassar rörelsen utefter fogen.

5.5 MALGENERATOR - ROBOTENS VÄRLDSBILD

Den information som skall behandlas från sensorerna ger dels en intern, dels en extern bild av hur robotsystemet arbetar och hur processen utförs.

Den interna bilden ger möjlighet att kontrollera att robotsystemet utför sitt arbete inom de toleransområden som bestäms av dess egenskaper. Detta innebär att den kalibrerade sensorinformationen jämförs med tillåtna värden på systemet som temperatur i styrsåpet, strömmar till drivdon, arbetsområde och eftersläpning av de enskilda rörelseaxlarna.

Den externa bilden ger möjlighet att utifrån resultatet av processen förändra robotsystemets beteende för att erhålla ett önskat resultat. Vid fogföljning gäller då speciellt att kunna känna igen situationer som kräver olika strategi vid val av återkopplingsinformation till rörelsegenereringen. Exempel på detta är sökning och ingång till fog vid svetsstart, rörelser vid svetslut, följdning av raka fogar och kurvformade fogar (beroende på hur sensorn är placerad och det mätfel som uppstår).

Mellan det interna och externa informationsflödet finns en koppling som gör att det kan uppstå en ömsesidig påverkan. På robotnära nivå kan detta innebära att robotsystemets rörelseaxlar kommer utanför sina arbetsområden vid fogföljning. Genom den programässiga kontroll som utförs av robotsystemets egna rörelser kommer detta att upptäckas och rörelsen samt processen kan avbrytas på ett kontrollerat sätt. Vid rörelser utan sensoråterkoppling kan en rörelse som resulterar i överskridande av arbetsområdet begränsas till att vara inom arbetsområdet beroende på den tolerans som tillåts för rörelsen. Först då denna överskrids kommer rörelsen att avbrytas. Man erhåller genom detta förfarande rörelser som går utanför robotens arbetsområde utan att rörelsen avbryts. Nackdelen är att det ger en ökad risk för oväntade kollisioner med omgivningen på grund av en oväntad rörelse. Detta är emellertid ett problem som ofta uppstår vid programmering och intrimning av robotar. Anledningen till att funktionen har använts är att systemet inte behöver stoppa på grund av en felprogrammerad punkt.

Bestämningen av robotsystemets arbetsområde bestäms av den mekaniska strukturens rörelsemöjlighet. I baskoordinatsystemet eller på objektnivå finns dessutom möjlighet att ange förbjudna områden. Detta är speciellt värdefullt vid bestämning och kontroll av till- och frånrörelser vid bågsvetsning.

Vid arbetsprocesser i industrimiljö beror arbetets utförande på ett mycket stort antal parametrar och variabler. Detta gör det mycket svårt att ställa upp styralgoritmer för en arbetsprocess utom i mycket begränsade fall och med väl definierade förutsättningar. Om man vid en studie av en arbetsprocess kan plocka ut ett relativt litet antal typfall och styralgoritmer som täcker de förutsättningar som normalt förekommer i arbetsprocessen, kan man utifrån sensorinformationen skapa återkopplingsdata som för olika situationer

gör att robotsystemet kan uppfylla målet.

I det presenterade styrsystemet har återkoppling av rörelsen simulerats genom att enligt metodiken ovan utnyttja olika strategier vid ingång till svetsfog och för fogar med olika kurvform. Vid detta arbete kan man konstatera de fördelar som ett modulariserat system ger beträffande systemuppbyggnad och funktion enligt avsnitt 4.1.2. För att bedöma hur systemet uppträder tillsammans med mekaniken krävs praktiska prov på grund av de svårigheter som uppstår i att bygga upp exakta dynamiska modeller som beskriver robotsystemets rörelser.

5.6 RESULTAT AV SYSTEMET FUNKTIONER

Vid uppbyggnaden av styrsystemet har de olika systemfunktionerna definierats som moduler för att erhålla klart avgränsade delar. Kärnan i systemet består av realtids styrsystemmodulen och dess undermoduler. Beräkningsschemat för rörelsegenereringen under en klockcykel blir för de tre första nivåerna:

Indata	Modul	Utdata
Rörelsetyp, start, slut, hast	Rörelse nivå 3	X, Y, Z, τ, ϕ
X, Y, Z, τ, ϕ	Rörelse nivå 2	$\phi, \theta, \alpha, \tau, \phi$
$\phi, \theta, \alpha, \tau, \phi$	Rörelse nivå 1	$(\phi, \theta, \alpha, \tau, \phi)_{inc}$
$(\phi, \theta, \alpha, \tau, \phi)_{inc}$	I/O	

På motsvarande sätt samlar sensorsystemet in information från processen och utför beräkningar så att den blir relaterad till rörelsebeskrivningen på olika nivåer i hierarkin. Målgeneratoren erhåller information dels från sensorsystemet, dels från rörelsegenereringen samt indataskommandon (t ex val av sensor eller styralgorithm).

Klockcykeln i det presenterade systemet är 10 ms. Vilka krav kommer detta att ställa på beräkningarna? Följande uppställning ger en indikation på hur antalet operationer fördelar sig på olika nivåer i rörelsegenereringen.

Nivå	Mult/Div	Add/Subtr	Kvadratrot	Sin/Cos/Arctan
1	30	19	5	4
2	26	19	4	15
3a	13	15	0	0
3b	116	50	0	4
4*	160	65	2	10

*) antal operationer/sökning i en sökprocess

Man får i praktiken räkna med flyttalsberäkningar och en dubblering av antalet operationer på grund av kontroll av numeriska värden, då singulariteter eller överskridna gränsvärden kan uppstå i beräkningarna. Vid återföring av information via sensorsystemet krävs motsvarande beräkningar i respektive nivå. Nivå 3a representerar linjär interpolering av en robotrörelse med TCP relaterad till roboten (stillastående objekt). Vid initieringen av en linjärrörelse krävs dock flera operationer än vad som angivits. Dessa kan emellertid förberäknas vid systemstart. Nivå 3b anger antalet ytterliggare operationer som krävs i nivå 3 för att åstadkomma linjär interpolering med TCP relaterat till rörligt objekt på en lägesställare typ ORBIT 500. Nivå 4 anger antalet operationer som krävs i varje sökning för att söka sig fram till position och orientering av lägesställare och robot. För en kort, rak fog, där hela rörelsen ligger inom robotens arbetsområde, krävs minst två sökprocesser för att få fram rörelsen (start- och slutpunkt). Antalet sökningar i varje sökprocess kan räknas i hundratal och man ser att antalet operationer för de högre funktionerna blir mycket stort jämfört med de lägre nivåerna.

Programvaran till det presenterade systemet har byggts upp i en enda dator. Den hierarkiska strukturen medger emellertid en parallell behandling av många funktioner. Man ser enligt uppställningen av antalet operationer och systemets uppläggning att det finns naturliga snitt att dela upp beräkningarna på flera processorer. Enligt beskrivningen av systemet tidigare består det av funktionellt avgränsade delar, rörelsegenerering, sensorsystem och målgenerator. En uppdelning av beräkningarna till dessa funktioner ger svårigheter vid dimensionering av beräkningskapaciteten, eftersom den vid de högre nivåerna är föränderlig genom ändringar i mekanisk struktur eller beräkningsrutiner. Om man ser till uppbyggnaden av systemet, där de lägre nivåerna

definierats först och sedan en succesiv påbyggnad av nya funktioner, kan en uppdelning av beräkningarna på olika processorer göras enligt följande:

- Nivå 1 och 2. Till största delen robotberoende rörelsegenerering och återkoppling. Informationen berör hur de olika rörelseaxlarna skall röra sig eller hur robotsystemets enskilda delar arbetar.
- Nivå 3a. Rörelser med TCP relaterat till roboten (stillastående objekt). Beräkningarna för rörelsen är relativt enkla. Återkopplingsinformationen och hur den skall utnyttjas kan skifta beroende på dess funktion.
- Nivå 3b. Rörelseberäkningen är krävande och unik för varje typ av lägesställare.
- Nivå 4 och högre. Mycket stort antal operationer. I och med att beräkningarna berör en svetsrörelse eller objekt i taget, ligger realtidskravet betydligt lägre än vad som gäller för nivå 1, 2 och 3.

Inledningsvis presenterades en kravspecifikation som beskrev de egenskaper som ett robotsystem för bågsvetsning skall ha för att kunna användas vid småserietillverkning. Dessa krav och hur de löses i det presenterade systemet beskrivs nedan:

- a) Systemet skall kunna hämta och processa sensordata från omvärlden och utifrån dessa data ändra robotens arbetssätt på ett målinriktat sätt så att eventuella fel minimeras till en tillåten toleransnivå.

Genom den starkt modulariserade uppbyggnaden i parallella hierarkier för rörelsen och sensoråterkopplingen underlättas och förenklas hanteringen och behandlingen av sensorinformation.

- b) Systemet skall kunna programmeras med ett målinriktat och kraftfullt kommandospråk på olika nivåer så att programmeringstiden vid robotsystemet blir så kort som möjligt.

Genom högnivåkommandon, där rörelsen utförs relativt objektet istället för ett till roboten fixerat koordinatsystem, kommer programmeringstiden vid roboten att reduceras väsentligt för ett flermanipulatorsystem jämfört med konventionell teknik. Planerande funktioner som bestämmer manipulatorernas positioner och rörelser utifrån objekt och process innebär en ytterliggare reduktion, där själva programmeringsarbetet består i att kalibrera eller korrigera förberäknade svetsrörelser samt eventuellt definiera hur till- och frånrörelserna skall utföras.

- c) Systemet skall kunna kommunicera med andra system och externa databaser som hjälp vid programmeringen och val av processdata.

Vid användning av högre funktioner som planerar och beräknar svetsrörelser får man en naturlig koppling till användning av databaser som beskriver objektet. Exempel på sådan databas är produktinformationen från ett CAD/CAM-system som kan representera objekt med ytor.

- d) Systemet skall vara enkelt utbyggbart för nya funktioner och ändrade arbetsförhållanden.

Genom den modulariserade uppbyggnaden i flera nivåer erhålles förutsättningar för att ändra eller lägga till funktioner utan stora arbetsinsatser. Eftersom systemet byggs upp från ett robotorienterat synsätt till högre funktioner som är av objekt- eller processororienterad natur, kommer förändringarna i systemet mestadels att beröra de högre funktionerna som berör objekt och process.

- e) Styrsystemet till robotsystemet skall i största utsträckning vara oberoende av olika mekaniska strukturer (robot, lägesställare) och enkelt att anpassa till dessa.

Den strukturella uppbyggnaden av systemet innebär ingen lösning till en speciell mekanisk struktur hos robotsystemet. De förändringar som behöver göras vid ändrad mekanik är ändring av transformationrutiner och rörelseplanering i de högre funktionerna.

- f) Robotsystemet skall automatiskt kunna styra svetsprocessen till ett fullgott resultat.

Problematiken kräver ytterliggare forskningsinsatser för att bestämma hur styrbara svetsparametrar och rörelsen påverkar varandra till ett önskat resultat. Det presenterade systemet ger emellertid förutsättningar för att utföra och beräkna rörelser och återkopplingar i realtid för bågsvetsning.

5.7 REFERENSER

- 5.1 G. Bolmsjö, Sensorinteraktiva Styrssystem för industrirobotar, Mekanisk Teknologi, LTH, Lund 1984 (del B i STU-rapport 82-4117, Flexibelt Robotsystem för Montering och Verktygsbärande Uppgifter).
- 5.2 RB2 Servicekurs Del 1 Beskrivningar, IRb6,60/S1, ASEA Robotics inc.
- 5.3 J. Bollinger and N. Duffie, Computer Algorithms for High Speed Continuous-Path Robot Manipulation, Annals of the CIRP Vol 28/1/1979.
- 5.4 J. H. Ahlberg, E. N. Nilson, J. L. Walsh, The Theory of Splines and Their Applications, Academic Press inc. (New York, USA), 1967.
- 5.5 P. G. Ranky and C. Y. Ho, Robot Modelling. Control and Applications With Software, IFS(Publ) Ltd, UK, 1985.
- 5.6 F. Matsuda, M. Ushio and Y. Tanaka, Metal Transfer Characteristics in Pulsed GMA Welding, Trans of JWRI Vol 12/1 1983.

6. ROBOTSYSTEMETS MÖJLIGHETER OCH BEGRÄNSNINGAR VID BÄG-SVETSNING

För att en robotstation för bågsvetsning skall ge ett förväntat resultat krävs att man på förhand definierar vad som skall svetsas och vilka resultat man vill uppnå.

Om man eftersträvar en säker och tillförlitlig produktion, bör man söka sig till sådana arbetsområden i processen som uppvisar minst känslighet i variationer i svetsparametrarna. Detta förhållande förstärks vid småserier, där man vill undvika tidsödande intrimningsarbete för att säkerställa ett önskat resultat. Vid robotsvetsning i större tillverkningssystem är det dessutom en fördel att kunna hålla en jämn produktion med få avbrott. För att kunna utföra en objekt- eller processororienterad programmering av rörelserna krävs kunskap om svetsparametrarnas relationer till varandra och hur dessa påverkar resultatet. De parametrar som avses i begreppet svetsparametrar är följande, se bild 6.1:

- 1) fogtyp, grundmaterial, svetstråd, gasblandning
- 2) ström, spänning, trådmätning, svetspistolens lutning (två vinklar) och läge (sidoförskjutning), fogens lutning (två vinklar), svetshastighet, utstick, svetsföljd
- 3) gasmängd, kylning

Parametrarna i grupp ett kan för ett objekt i praktiken anses vara låsta. Val av fogtyp och grundmaterial är inbyggda i konstruktionen. Svetstråd och gasblandning kan i princip ändras genom byte av svetspistol. Detta leder emellertid till tidsförluster och ökad haveririsk genom flera ingående komponenter och bör om möjligt undvikas.

Parametrarna i grupp två är de styrbara parametrarna. Mellan ström och trådmätning finns en koppling som gör att en vald ström ger en bestämd trådmätning. Parametrarna i strömkällan är då ström och spänning samt den bågtyp som kan åstadkommas genom att styra dessa (t ex SPM). De övriga är

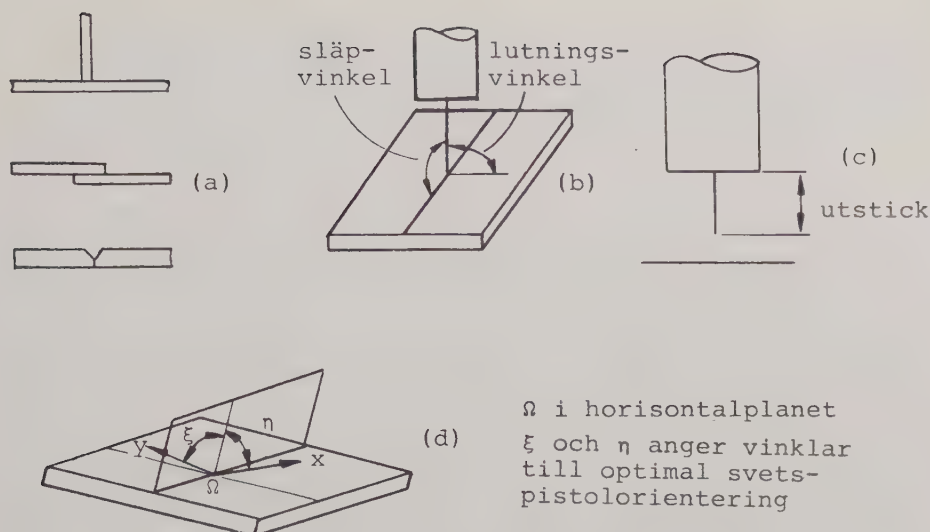


Bild 6.1. Svetsparametrar. Fogtyp (a), Svetspistolens lutning (b), fogens lutning (c), utstick (d).

parametrar som till största delen bestäms av manipulatorernas rörelser. I vissa fall ställer dock objektet begränsningar genom reducerad åtkomlighet eller olämplig orientering av svetsfogarna. Svetsföljden kan idag inte bestämmas genom automatiska algoritmer. Detta beror på den frihetsgrad som finns i utformningen av objekt och de problem som måste beaktas beträffande spänningar och deformationer genom det tillförda värmets. De övriga parametrarna (svetspistolens orientering och läge, fogens orientering, svetshastighet och utstick) kan beräknas och kontrolleras genom de högre funktionerna beskrivna i kapitel 5.

Parametrarna i grupp tre är till funktionen till/från, dvs antingen tillräcklig gasmängd och kylning eller ej. I princip kan dock styrning av gasmängden utnyttjas för att spara gas. Detta är emellertid kopplat till fogtyp och gasflödet genom svetspistolen.

Man kan konstatera att val av och kopplingen mellan svetsparametrarna ovan inte är klarlagda så att de automatiskt kan bestämmas vid robotsvetsning. Vid generering av svetsrörelser skall parametrarna bestämmas till värde och

toleransområde. För att säkerställa funktionen i processen bör då parametrarna väljas så att de toleransområden som gäller för rörelsen kan maximeras. Avsikten med detta är att

- minimera det kritiska toleransområdet på objekt och fixtur
- minimera behovet av sensorer som skall kompensera för fel i svetsrörelsen relativt objektet
- öka toleransområdet för manipulatorernas rörelseområde. Detta är viktigt då man genom objektets utformning inte kan räkna med att generera en optimal svetsrörelse

Speciellt den sista punkten ovan är viktig vid genereringen av rörelsen. Vid automatisk generering av rörelser utnyttjas algoritmer som ger möjliga rörelseområden med en definierad tolerans relativt en optimal rörelse. Vid svetsning av generella objekt måste man dessutom anta att man måste utnyttja tillåten rörelsetolerans. Detta beror på lägesställarens begränsade orienteringsförmåga, robotens begränsade arbetsområde samt eventuella åtkomlighetsproblem på objektet. I och med detta inser man, att en kontrollerad tolerans på objekt och fixtur ger möjlighet att bättre definiera och öka manipulatorernas rörelsetolerans. Detta gäller även behovet av att använda fogföljarsensorer. Om rörelsetoleranserna skall kunna bestämmas, måste robotsystemets arbetsområde definieras. Enligt [6.1] kan svetsparametrarnas relationer till varandra erfarenhetsmässigt och i praktisk produktionssvetsning beskrivas enligt följande:

- objekt i stål är mindre känsliga än objekt i andra materialtyper
- grövre plåttjocklekar har större toleransområden än mindre tjocklekar
- svetstråden bör kunna fixeras till 1.0 eller 1.2 mm samt 1.6 mm rörtråd vid grövre dimensioner
- skyddsgasen är i praktiken normalt 80% Ar och 20% CO₂. Skyddsgas bör väljas så att sprutet minimeras. Detta kan innebära andra sammansättningar än vad som används idag
- bågtyperna som används är kortbåge, SPM och äkta spraybåge

- fogtyperna är i första hand kälfog och överlappsfog. Undersökning av stumfogens svetsbarhet kräver noggrannare studier bl a avseende betydelsen av pendling och pulsning. Hörnfog ger risk för genombränning och bör undvikas. Spalten i fogarna bör kunna ligga i området 0-1.5 mm, beroende på plåttjocklek

I det presenterade robotsystemet används en lägesställare som hanterar objektet. I praktisk produktion innebär detta småserietillverkning av operat med begränsad massa (upp till ca 500 kg). Om man tar hänsyn till detta, kan man fastställa plåtdimensionerna inom området 4-20 mm. Använda material är svetskonstruktionsstål och behöver ej förvärmas. Utifrån detta kan rörelsernas parametervärden och dess toleransområden ges enligt följande [6.1]:

Fogorienteringen bör vara horisontell och svetspistolen bör ligga i vertikallinjen. Erfarenhetsmässigt är det en fördel att svetsa i 10° nerförsbacke, men som normalvärde bör fogen orienteras horisontellt $\pm 10^\circ$. Om inte detta kan uppnås, bör man söka vrida fogen kring dess längsaxel, dock inte mera än att svetspistolen får en orientering relativt vertikallinjen $\pm 90^\circ$. I sista hand kan fogen lutas mer än 10° i backe. Svetsning underupp rekommenderas ej p g a risk för sprut och igensättning av munstycket i svetspistolen.

Svetshastigheten är kopplad till fogorienteringen, så att horisontell fog tillåter en högre hastighet än lutande fog med samma kvalitetskrav. En övre gräns för hastigheten sätts genom ökat sprut och igensättning av munstycket. Vid lutning av svetsfogen bör på motsvarande sätt hastigheten och strömmen avpassas.

Svetspistolens orientering relativt fogen bör vara vinkelrät med en uppskattad tolerans på $\pm 5^\circ$ i släpvinkel och lutningsvinkel. Vid denna uppskattning antas lika plåtdimensioner i grundmaterialet.

Utsticket kan som ett normalvärde väljas till 10-12 mm. Toleransen är beroende av bågtyp och plåttjocklek. Om man går upp i plåttjocklek till 8 mm och använder äkta spraybåge, bör toleransen ligga på ± 3 mm. En gräns för toleransen sätts dessutom av trådens krökning, som vid stora utstick kan försämra sidoläget.

Sidolägets tolerans anges erfarenhetsmässigt till ± 0.5 mm vid MIG/MAG-svetsning. Toleransens storlek är kopplad till plåttjocklek och fogtyp. För generering av svetsrörelser har inte denna tolerans någon praktisk betydelse, utan anger istället krav som ställs på noggrannhet i fogberedning, fixtur och manipulatorer.

Bestämning av svetsföljden ingår i operationsberedningen och är beroende av bakomliggande kunskaper om hur värmespänningar uppstår och byggs in vid svetsning, fixturteknik samt hur förflyttningsrörelserna mellan svetsfogarna kan optimeras. Denna parameter är komplex och kräver mycket omfattande studier innan en automatisering av svetsföljden kan utföras på komplicerade objekt.

Enligt ovan ser man hur värdefullt det är att kunna ange gränserna för hur ett beskrivet robotsystem skall arbeta i processen. Här ligger en möjlighet till automatisk beredning och automatisk programmering av bågsvetsning med robotar.

6.1 REFERENSER

- 6.1 Personliga samtal med B. Pekkari, B. Torstensson, M. Berg, P-A. Larsson, K-E. Knipström, ESAB AB.

7. DET PRESENTERADE SYSTEMETS MÖJLIGHETER ATT REALISERAS FÖR PRAKTISK DRIFT

Det presenterade robotsystemet för bågsvetsning vid småserier beskriver en uppbyggnad som underlättar hantering av sensorinformation samt planering och generering av svetsrörelser anpassade till objekt och process. Genom detta erhålles

- reducerad programmeringstid vid robotstationen genom utnyttjande av objektprogrammering i tvåmanipulatorsystem. Vid processprogrammeringen sker en ytterliggare reducereing genom automatisk planering och generering av svetsrörelserna
- minskad olycksrisk vid robotstationen genom minskad programmeringstid. Olyckor vid robotar kan inträffa då människor måste komma in i robotens arbetsområde. Då programmeringstiden normalt är lång vid bågsvetsning, bör en reducereing av denna även innebära en märkbar reducereing av olycksfrekvensen vid robotar
- den presenterade tekniken att utföra processprogrammering ger möjlighet att analysera om ett objekt är lämpligt att svetsa i en utvald robotstation
- genom processprogrammering, där svetsrörelserna genereras automatiskt, presenteras ett underlag och teknik som möjliggör off line programmering av tvåmanipulatorsystem för bågsvetsning
- uppbyggnaden av styrsystemet underlättar återkoppling av sensorinformation i realtid på alla nivåer. Systemet är genom uppbyggnaden enkelt att anpassa till olika sensorer och återkopplingsalgoritmer

En automatisk generering av svetsrörelser innebär, att manipulatorernas rörelser beräknas numeriskt och överförs till robotsystemet som vid off line programmering. Detta medför, att det kan uppstå noggrannhetsproblem, speciellt vid tvåmanipulatorsystem och mellan olika robotstationer. För att praktiskt använda denna beräkningsteknik krävs antingen en manuell korrigering av de beräknade rörelserna eller en automatisk kalibrering av rörelserna. Den automatiska kalibreringen av rörelserna kan ske

- i överordnad dator genom en kalibreringsfunktion som på förhand är uppmänt och unik för varje robotstation
- i överordnad dator genom ett uppmättningsförfarande som utförs varje gång en ny serie startar. Genom att göra denna mätning mot fixturen erhålls samtidigt en korrigerig av fel som kan hänföras till orientering/positionering av fixtur och objekt
- i robotsystemet genom fogsökning och/eller fogföljning. Tekniken tar även hänsyn till fel i objekt och fogar och deformationer som uppstår under processen. Tekniken ger emellertid tidsförluster genom den fogsökning som sker vid varje fog jämfört med de övriga metoderna ovan

De ovan beskrivna kalibreringsteknikerna kan kombineras för att få önskad effekt. En generering av svetsrörelserna i överordnad dator ger en naturlig koppling till CAD/CAM. Genom utnyttjande av CAD-tekniken vid definiering av objektet och dess svetsfogar underlättas indata till processprogrammeringen, bestämning av till/från-rörelser samt simulering av rörelser och kollisionstest.

8. FRAMTIDA FORSKNING

Vid fortsatt utveckling och automatisering av robotiserad bågsvetsning krävs forskningsinsatser för att klargöra samspelet mellan robotsystem och svetsprocess. Bland dessa kan nämnas

- automatiskt val av svetsparametrar. Detta innebär att samspelet mellan svetsparametrarna måste bestämmas så att val kan göras t ex för bestämd svetskvalité, erhållna rörelser och svetsfölj
- fixturteknik och fogförberedning vid robotsvetsning och småserier kräver speciell metodik för att säkerställa toleranskrav och minska behovet av häftsvetsning och/eller sensorer
- sensorteknik och återkoppling av styrdata kräver insatser för att få en praktisk tillämpning vid varierade situationer som fogtyp och häftsvetsning
- automatisk kalibrering och korrigering av program. För att lösa de noggrannhetsproblem som uppstår vid beräkning av rörelser i överordnad dator krävs rutiner som kompenserar för fel som uppstår p g a toleranser och förändringar i mekanik, last, objekt, etc
- användning av AI-teknik för att lösa de problem som uppstår vid komplexa svetsproblem. Detta kan gälla analys av geometrier eller fogtyper, andra material (t ex aluminium, rostfritt och titan) samt hur svetsföljden påverkar spänningsuppbyggnaden i objektet

Av ovanstående framgår att det krävs en helhetssyn av den totala problematiken för att programmering och beredning skall kunna utföras automatiskt. Detta innebär att svetsprocessen studeras med hänsyn till de villkor som gäller för att erhålla repeterbara och stabila resultat samt att rörelserna för dessa är anpassade för robotsvetsning. Robotsystem för bågsvetsning innehåller ofta externa rörelseaxlar. I sådana system krävs algoritmer som utifrån processkrav och kollisionstester automatiskt kan planera och generera rörelser. I de fall då toleranserna i rörelse eller objekt är kritiska, krävs en hantering av sensorinformation som gör det möjligt att anpassa rörelsen efter det verkliga objektet och processens toleransgränser.

